Humboldt-Universität zu Berlin

DISSERTATION

Privacy trade-offs in web-based services

Zur Erlangung des akademischen Grades
doctor rerum politicarum
(Doktor der Wirtschaftswissenschaft)

eingereicht an der
Wirtschafswissenschaftlichen Fakultät
der Humboldt-Universität zu Berlin

von

Herrn Diplom-Wirtschaftsingenieur Claus Boyens
geboren am 17.9.1975 in Kiel

Präsident der Humboldt-Universität zu Berlin:
Prof. Dr. Jürgen Mlynek

Dekan der Wirtschaftswissenschaftlichen Fakultät:
Prof. Dr. Joachim Schwalbach

Gutachter: 1. Prof. Oliver Günther, Ph.D.

2. Prof. Ramayya Krishnan, Ph.D.

eingereicht: 26. August 2004
Datum der Promotion: 15. Dezember 2004

Abstract

Recent developments in networking and storage technology have led to the dissemination of information over many different sources such as personal computers or corporate and public databases. As these information sources are often distributed and heterogeneous, effective tools for data collection and integration have been developed in parallel. These tools are employed e.g. in library search catalogues or in Internet search engines to facilitate information search over a wide range of different information sources.

In more sensitive application areas however, the privacy of the data holders can be compromised. In medical disease research for example, scientists collect and analyze patient data for epidemiological characterizations and for the construction of predictive models. Whereas the medical researchers need patient data at the highest level of detail, patients are only willing to provide data when their privacy is guaranteed. This conflict of interest between the data holders and the users occurs in many different settings, for example in the use of web-based services that require confidential input data such as financial or tax data. The more accurate and rich the provided private information, the higher the quality of the provided service. Not all data holders are aware of this trade-off and for lack of knowledge tend to the extremes, i.e. provide no data or provide it all.

This thesis explores the borderline between the competing interests of data holders and service users. In particular, we investigate the technical opportunities to model and describe this borderline. These techniques allow the two opposing parties to express their preferences and to settle the conflict with a solution that is satisfactory to both. The specific contributions of this thesis are the following:

- *Privacy classification of service architectures*
  We present a privacy classification of different service architectures after the number of involved parties and the reactivity of the data provision. For each class, we provide examples of practical applications and explain their relevance by discussing preceding cases of real-world privacy violations.
- *Design, analysis and implementation of an encryption-based service architecture in an untrusted two-party environment*
  We analyze the foundations of trust in web-based services and point out cases where trust in the service provider alone is not enough e.g. for legal requirements. For these cases, we derive a new privacy-preserving architecture that is based on an adapted homomorphic encryption algorithm. We map important database and arithmetic operations from plain data to encrypted data, and we present sample services that can be carried out within the framework.

- *Design, analysis and implementation of an aggregation-based service architecture in an untrusted three-party environment*

  Using a privacy-compromising health report as a running example through the thesis, we show how mathematical programming can be used to derive tight intervals for confidential data fields from non-critical aggregated data. We propose a new class of privacy mediators that settle the conflict between data holders and service users. A core component is the "audit & aggregate" methodology that detects and limits this kind of disclosure called interval inference.

- *Quantification of the privacy trade-off and implications for electronic commerce and public policy*

  We analyze several frameworks to quantify the trade-off between data holders and service users. We also discuss the implications of this trade-off for electronic commerce and public policy.

To summarize, this thesis aims to (a) increase data holders' and service users' awareness of the privacy conflict, (b) to provide a framework to model the trade-off and (c) to develop methods that can settle the conflict to both parties' satisfaction.

Keywords:

Privacy, Security, Confidentiality, Encryption, Aggregation, Electronic Commerce, Service Architecture

Zusammenfassung

Rapide Fortschritte in der Netzwerk- und Speichertechnologie haben dazu geführt, dass Informationen über viele verschiedene Quellen wie z.B. Personal Computer oder Datenbanken verstreut sind. Weil diese Informationen oft auch sehr heterogen sind, wurde gleichzeitig die Entwicklung effektiver Softwaretechniken zur Datensammlung und -integration vorangetrieben. Diese werden beispielsweise in Online-Katalogen von Bibliotheken oder in Internetsuchmaschinen eingesetzt und ermöglichen eine breitgefächerte Suche von Informationen unterschiedlichster Art und Herkunft.

In sensiblen Anwendungsgebieten kann der Einsatz solcher Techniken aber zu einer Gefährdung der Privatsphäre der Datenhalter führen. Bei der Erforschung häufig auftretender Krankheiten beispielsweise sammeln und analysieren Wissenschaftler Patientendaten, um Muster mit hohem Erkrankungspotenzial zu erkennen. Dazu werden von den Forschern möglichst präzise und vollständige Daten benötigt. Der Patient hat dagegen großes Interesse am Schutz seiner persönlichen Daten. Dieser Interessenkonflikt zwischen Datenhaltern und Nutzern tritt auch in anderen Konstellationen wie beispielsweise in Internetdiensten auf, die die Eingabe von persönlichen Finanz- und Steuerdaten erfordern. Oft kann ein qualitativ höherwertiger Dienst angeboten werden, wenn persönliche Informationen preisgegeben werden. Über die hierzu notwendige Abwägung von Datenschutz und Dienstqualität sind sich nicht alle Datenhalter im Klaren und neigen zu Extremverhalten wie der Übermittlung aller persönlicher Daten oder gar keiner.

Diese Dissertation erforscht den Grenzbereich zwischen den scheinbar konträren Interessen von Datenhaltern und Dienstnutzern. Dabei werden insbesondere die technischen Möglichkeiten zur Modellierung und Beschreibung dieses Bereiches betrachtet. Die erarbeiteten Techniken sollen den beteiligten Parteien ermöglichen, den bestehenden Konflikt unter Einbeziehung ihrer Präferenzen zur beiderseitigen Zufriedenheit zu lösen. Die Beiträge dieser Dissertation sind im Einzelnen:

- *Eine Klassifizierung von Dienstarchitekturen im Hinblick auf Datenschutzprobleme*
  Verschiedene Dienstarchitekturen werden nach ihrer Datenschutzproblematik klassifiziert. Für jede Kategorie werden praktische Anwendungen erläutert.
- *Entwurf, Analyse und Implementierung einer verschlüsselungsbasierten Dienstarchitektur in einer nicht vertrauenswürdigen 2-Parteien-Umgebung*
  Es werden Gründe für Vertrauen von Datenhaltern in Anbieter von netzbasierten Diensten dargestellt. Für Fälle, in denen dieses Vertrauen alleine nicht ausreicht, wird eine Datenschutz garantierende Dienstarchitektur abgeleitet, die auf einem

modifizierten Verschlüsselungsalgorithmus basiert. Wichtige Datenbankoperationen und arithmetische Elemente werden auf die verschlüsselten Daten übertragen und in beispielhaften Diensten zum Einsatz gebracht.

- *Entwurf, Analyse und Implementierung einer aggregationsbasierten Dienstarchitektur in einer nicht vertrauenswürdigen 3-Parteien-Umgebung*
  Am Beispiel eines den Datenschutz verletzenden Gesundheitsberichts wird gezeigt, wie Methoden des Operations Research dazu eingesetzt werden können, aus veröffentlichten Statistiken enge Intervalle für vertrauliche numerische Daten abzuleiten ("Intervallinferenz"). Zur Lösung des Interessenkonflikts zwischen Datenhaltern und Dienstnutzern wird die Verwendung eines sogenannten Datenschutzmediators vorgeschlagen. Dessen Kernkomponente ist die "Audit & Aggregate" Methodologie, die das Auftreten von Intervallinferenz aufdecken und verhindern kann.
- *Quantifizierung der Datenschutzabwägungen und Schlussfolgerungen für den elektronischen Handel*
  Es werden verschiedene Ansätze zur Quantifizierung der Datenschutzabwägungen betrachtet und Schlussfolgerungen für den elektronischen Handel gezogen.

Zusammengefasst versucht diese Arbeit, (a) die Wahrnehmung von Datenhaltern und Dienstnutzern für den bestehenden Interessenkonflikt zu erhöhen, (b) einen Rahmen zur Modellierung der Datenschutzabwägungen bereitzustellen und (c) Methoden zu entwickeln, die den Interessenkonflikt zur beiderseitigen Zufriedenheit beilegen können.

Schlagworte:

Datenschutz, Sicherheit, Vertraulichkeit, Verschlüsselung, Aggregation, Elektronischer Handel, Dienstarchitekturen

Meinen Eltern

Acknowledgements

Writing this thesis would not have been feasible without the unfaltering support of my doctoral advisor Oliver Günther. I thank him for his guidance and for his advice throughout my years at the Institute of Information Systems.

I am grateful to Ramayya Krishnan and to Rema Padman for hosting me at Carnegie-Mellon University and for giving me valuable support and feedback for my research.

I would also like to thank the professors and my fellow Ph.D. students from the Berlin-Brandenburg Graduate School in Distributed Information Systems for valuable discussions and good spirit. I thank the Deutsche Forschungsgemeinschaft for the funding of my scholarship in the Graduate School (DFG grant no. GRK 316/3).

Thanks also go to my colleagues from the Institute of Information Systems for valuable discussions and feedback. I am particularly grateful to my office mates, Anett Kralisch and Max Teltzrow, for their good company.

Sarah Aerni from the University of Pittsburgh did the proof-reading for the entire thesis, which I deeply acknowledge here.

This thesis would not be in existence without the support of my friends and family. It is dedicated to my parents, Christian and Elisabeth Boyens, who have never ceased from supporting me in all imaginable ways.

Contents

Figures

Tables

Abbreviations

AMPL	Advanced Mathematical Programming Language
ASP	Application Service Provider
CADP	Critical Average Dual Price
DBMS	Database Management System
DR	Disclosure Risk
DS	Dissemination Strategy
DU	Data Utility
EPIC	Electronic Privacy Information Center
ERP	Enterprise Resource Planning
GUI	Graphical User Interface
HbA1c	Hemoglobin A1c
HIPAA	Healthcare Insurance Portability Accountability Act
HMO	Health Maintenance Organization
HR	Human Resources
HTML	Hypertext Markup Language
IC	Information Category
IL	Information Loss
LP	Linear Programming
NLP	Nonlinear Programming
P3P	Platform for Privacy Preferences
PH	Privacy Homomorphism
PIR	Private Information Retrieval
PRHI	Pittsburgh Regional Healthcare Initiative
RDP	Random Data Perturbation
SDC	Statistical Disclosure Control

SP	SERVICE PROVIDER
SQL	STRUCTURED QUERY LANGUAGE
SSL	SECURE SOCKET LAYER
TARE	TOTAL AVERAGE RELATIVE ERROR
VS	VALUE SUPPRESSION

1 Introduction

You have zero privacy anyway. Get over it.
(Scott McNealy, CEO of Sun Corp.)

1.1 Privacy trade-offs in web-based service environments

Recent developments in networking and storage technology have led to the distribution of information over many different sources such as personal computers or corporate and public databases. As these information sources are often distributed and heterogeneous, effective tools for data collection and integration have been developed in parallel. These tools are employed e.g. in library search catalogues or in Internet search engines to facilitate information search over a wide range of different information sources.

The collection and analysis of distributed data is useful and uncritical as long as the sources are publicly available, i.e. the data holders explicitly want to provide their data such as in personal homepages or in product catalogues. There are, however, more sensitive application areas such as cancer research. Medical researchers collect and analyze primary care data for epidemiological characterizations and for the construction of predictive models. Primary care data is created and recorded when a physician diagnoses, treats and medicates a patient and is of course strictly confidential. Necessary security measures include granting access to authorized people only and keeping the communication confidential. But even if these measures are in place when patient information is passed from physicians to researchers, many questions have not yet been answered with regard to the patients' privacy.

- Should individual patient records be anonymized; and if yes, which information should be removed?
- Should the researcher be allowed to share confidential information; and if yes, which information with whom?
- If a published research report contains aggregated confidential values of several patients, does it still respect each individual's privacy?
- If a researcher or his environment cannot entirely be trusted, are there ways to provide useful patient data while preventing misuse?

There are many instances where conflicts of interest arise between the data holders and

the users of the (possibly processed or modified) confidential data. We call these users "service users", where services also include the retrieval of the raw data as the simplest kind of service. In this case, the patients might agree on the benefits that a release of their data can have for curing their disease (if not for themselves, then for future generations), they have a natural interest in protecting their most confidential data from misuse (e.g., by the state, by their employer or by their insurance company). Principally, their aim is minimizing the risk of an invasion of their privacy by a third party. However, on the other hand the service users need confidential data at a maximal level of detail. Often the quality of the analysis improves with a more accurate and rich data set. Figure 1-1 sketches the contrasting aims of data holders and service users.

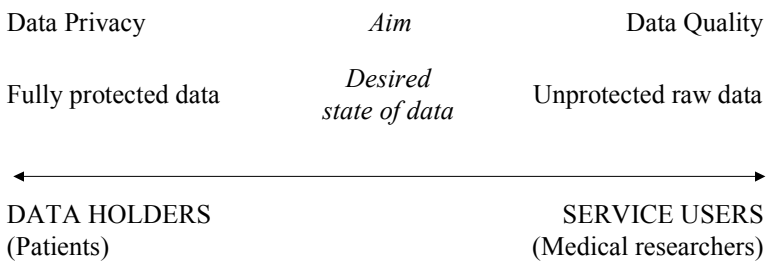


Figure 1-1: Different scope of data holders and service users

This thesis explores the borderline between the competing interests of data holders and service users. In particular, we investigate the technical opportunities to model and describe this borderline. Ideally, these techniques would allow each party to express their preferences and to settle the conflict with a solution that is satisfactory to both. Legal and organizational measures such as the one undertaken in [EU, 1995; EU, 2002] are important complements to technical measures, but are outside the scope of this work. For a detailed discussion, see [Ackerman, et al., 1999; Spiekermann, et al., 2001].

We will define a service-oriented framework to classify different types of privacy problems such as the cancer research case above. By *service*, we mean the storing or processing of confidential input data. In particular, we distinguish between a two-party case and a three-party case (see Figure 1-2).

The *two-party case* only involves the data holder and the service provider that uses and processes the input data to deliver the desired service result (see Figure 1-2 (a)). In the healthcare domain, a web-based service could be a personal health check based on input data such as age, gender, cholesterol, blood pressure and habits (e.g., smoking or sports activities). In this case, the data holder is the service user at the same time. The data holder may have the following concerns:

- The service provider forwards confidential data to an untrusted third party.
- The provider's database is subject to an external attack.
- The provider's staff is incompetent or bribed.
- Bankruptcy of the provider leads to uncertainty about data ownership.

We address these issues by presenting a new kind of service architecture. We make use of a specific class of encryption functions first introduced by [Rivest, et al., 1978a] that allows the data holder to use a service without actually transferring plain input data to the service provider. We elaborate in which contexts the architecture may be employed and give a computational analysis for particular services.

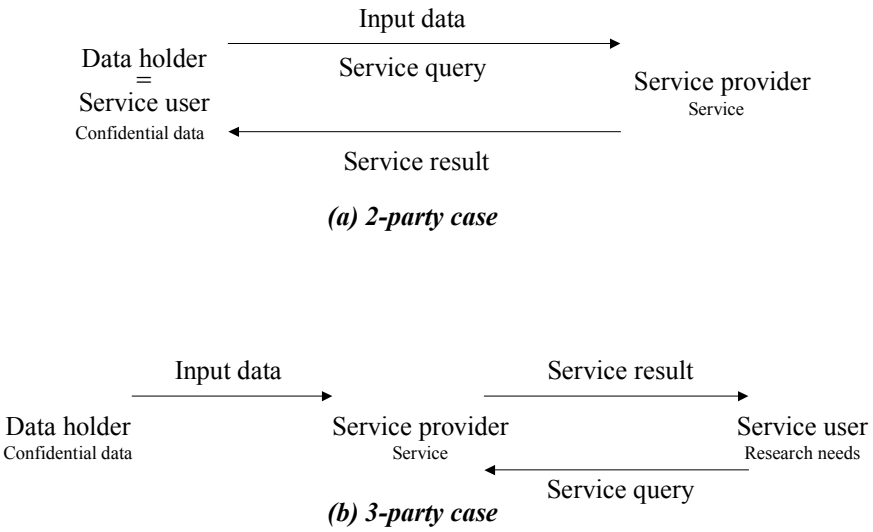


Figure 1-2: Two important cases of interaction between data holder and service user

In the *three-party case*, the data holder is not necessarily the same person as the service user. To give an example, the data holders may be the patients whose primary care data is given to a service provider. This service provider must ensure the confidentiality of patient data e.g. by anonymizing single data sets or by removing data attributes that are particularly critical before giving it to the actual service users, the medical researchers (see Figure 1-2(b)). The main privacy concerns on behalf of the data holders are the following in this case:

- Their confidential health information can be inferred from the data that is given to the public service users.
- The service provider cannot be trusted.
- Their confidential data is forwarded to an untrusted third party.

We present a mediator-based architecture based on [Wiederhold, et al., 1996] that

advocates these concerns on behalf of the data holder. We introduce an "audit & aggregate" methodology that protects the privacy of the data holders and, at the same time, maximizes the utility of the released data to prospective service users.

A commonality of the two- and the three-party-case is that the outcome of the service is directly related to the amount of private information provided by the data holders. The more accurate and rich the provided private information, the higher the quality of the provided service. Not all data holders are aware of this trade-off and for lack of knowledge tend to the extremes, i.e. provide no data or provide it all[1]. The objectives of this thesis are (a) to increase data holders' and service users' awareness of this issue, (b) to provide a framework to model the trade-off and (c) to develop methods that can settle the conflict to both parties' satisfaction.

1.2 Contributions

The specific contributions of this thesis are the following:

- *Privacy classification of service architectures*
  We present a classification of different service architectures with regard to privacy protection issues. For each class, we name examples of practical applications and explain the relevance by discussing preceding cases of real-world privacy violations.
- *Design, analysis and implementation of an encryption-based service architecture in an untrusted two-party environment*
  We analyze the foundations of trust in web-based services and point out cases where trust in the service provider is not enough e.g. for legal requirements. For these cases, we derive a new privacy-preserving architecture that is based on an adapted encryption algorithm. We map important database and arithmetic operations from plain data to encrypted data, and we present sample services that can be carried out within the framework. We evaluate our approach with regard to memory and performance requirements and we propose different implementation methods.
- *Design, analysis and implementation of an aggregation-based service architecture in an untrusted three-party environment*
  Based on a privacy-compromising health report as a running example, we show how

[1] According to [Ackerman et al., 1999], 17% of all online users are "privacy fundamentalists" who will not provide data to a web site even if privacy protection measures are in place. 27% are "marginally concerned" and generally willing to provide data to web sites without major concerns.

tight intervals for confidential data fields can be derived from non-critical aggregated data. We propose a new class of privacy mediators that settle the conflict between data holders and service users. A core component is the "audit & aggregate" methodology that detects and limits this kind of disclosure called interval inference.

- *Quantification of the privacy trade-off and implications for electronic commerce and public policy*
  We analyze several frameworks to quantify the trade-off between data holders and service users. We also discuss the implications of this trade-off for electronic commerce and public policy.

1.3 Structure of the thesis

The rest of this thesis is structured as follows.

Chapter 2 gives a classification of privacy problems in service architectures. We introduce the terminology and point to relevant related work. We also name occurrences of privacy issues in real-world information systems.

Chapter 3 discusses the case of the 2-party service architecture. We introduce a service provided by an Application Service Provider (ASP) as a running example. We then propose a privacy-preserving architecture that allows a service user to carry out a limited number off services on encrypted data.

Chapter 4 is concerned with the 3-party case. We use a regional healthcare initiative that collects, analyzes and disseminates chronic disease data as a running example. We derive several privacy problems and introduce a model that captures the case where tight bounds can be inferred on confidential values. We propose two new methods that limit this kind of privacy breach. These methods are implemented and compared to competing methods.

In Chapter 5, we extensively discuss the trade-off idea and ways to quantify it. We also discuss implications for electronic and public policy. We conclude in Chapter 6 with a summary and an outlook on future research. A sketch of this structure is captured in Figure 1-3.

Excerpts of Chapter 3 have been published in [Boyens and Fischmann, 2003; Boyens and Günther, 2002; Boyens and Günther, 2003]. Parts of Chapter 4 have been published in [Boyens and Günther, 2004; Boyens, et al., 2004; Boyens and Padman, 2003].

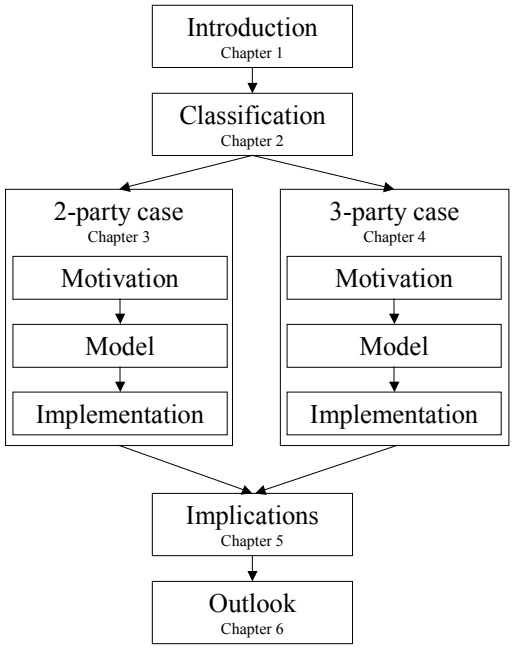


Figure 1-3: Structure of the dissertation

2 A classification of privacy issues in service architectures

The human animal needs a freedom seldom mentioned, freedom from intrusion. He needs a little privacy.
(Phyllis McGinley, American writer)

2.1 Definitions and terminology

2.1.1 Web-based services

A service that is provided via any kind of wired or mobile network is called *net-based service* [Tamm and Günther, 2004]. When these services are provided using technologies recommended by the World Wide Web Consortium[2] (such as HTTP [W3C, 1997]) and using the underlying transport/network protocol TCP/IP [DARPA, 1981a; DARPA, 1981b] , we speak of *web-based services*. Net-based services that do not follow these standards include mobile services and services that are based on proprietary technologies (such as internal company networks). Note that because of their relative novelty, no unambiguous definition has been established yet. For reasons of simplicity, we will from now on use the terms net-based and web-based services synonymously and denominate them as the latter.

An important subtype of web-based services are *web services* who are defined as "software systems identified by uniform resource identifiers, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols" [W3C, 2004]. These services do not require a graphical user interface and are mainly used to improve the communcation and interoperability between applications.

Complementary to this technological definition, web-based services can also be classified

[2] www.w3.org

following their business model. For example, an *Application Service Provider (ASP)* "deploys, hosts and manages access to a packaged application to multiple parties from a centrally managed facility. The applications are delivered over networks on a subscription basis. This delivery model speeds implementation, minimizes the expenses and risks incurred across the application life cycle, and overcomes the chronic shortage of qualified technical personnel available in-house" [IDC, 1999]. Usually, the ASP charges a flat fee per user from its client. We will refer to this business model further in Section 3.1.

2.1.2 Privacy issues

For the purposes of this thesis, we speak of a (web-based) service *S* when a service provider *SP* stores, modifies, processes, publishes or forwards some confidential input data *D* given by the data holder (see Figure 2-1). We assume that the service provider creates a service result *S*(*D*) that is either returned to the data holder (2-party case) or forwarded to an authorized third party (3-party case). In the 2-party-case, the data holder is also the service user, see the solid service line in Figure 2-1. In the 3-party-case, this is not necessarily the case, as an external third party is involved, as depicted by the dashed line in Figure 2-1.

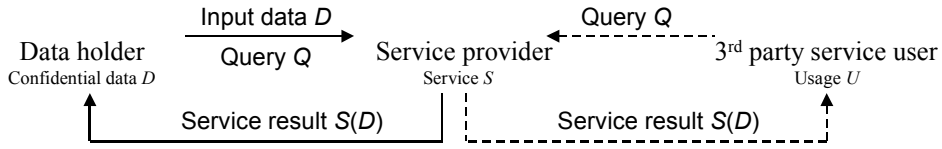


Figure 2-1: Input data and provided service

The input data *D* can be delivered *reactively*, e.g. by explicitly specifying weight and blood pressure for an online health check. Or it can be delivered *non-reactively / passively*, e.g. when a physician forwards patient data to a health service institution for research purposes.

Roles of the service provider include online stores (www.amazon.com), online service providers (such as financial services, e.g. www.citibank.com), data mining providers (e.g. www.datashaping.com) or regional health initiatives (e.g. www.prhi.org).

Exemplary roles of the third party include direct marketing companies, medical researchers, patients or yet another service provider.

Based on the data holder's trust level of trust towards the service provider and towards the

third party, we can distinguish four different privacy constellations. Table 2-1 illustrates this. For a detailed discussion of what influences trust in web-based services, see [Jarvenpaa, et al., 2000][3].

		TRUST IN 3rd PARTY	
		Yes (or 3rd party not existent)	No
TRUST IN SERVICE PROVIDER	Yes	**Uncritical**	**Transform / protect *S(D)*** **Chapter 4**
	No	**Transform / protect *D*** **Chapter 3**	**Transform / protect *D* and *S(D)*** **Chapter 4.1.4**

Table 2-1: The four privacy constellations depending on the data holder's trust

The straight-forward case is when trust exists both towards the service provider and towards the 3rd party. Privacy is not an issue.

When there is no third party or the third party can be trusted, the data holder only has to make sure that his confidential data *D* is protected against potential misuse at the service provider's site. This case usually applies for the use of online services that require the input of confidential data such as income data or personal health data, and is explained in Chapter 3.

Things become more difficult if an untrusted third party comes in. This may be the case in the sketched example of a regional health initiative that collects and analyzes primary care data (this is the service *S*) and distributes health reports to their community (the 3rd party that is not necessarily trusted). This case is addressed in Chapter 4.

The most delicate case occurs when neither the service provider nor the third party can be trusted. The opportunities to deliver useful service results in this case are very limited. Consider e.g. online voting [Asonov, et al., 2001]. Confidential data (the vote) is passed to the service provider (the state or another official voting institution) who aggregates the votes and publishes the election result to the public (i.e., the 3rd party). A voter would want to keep his vote secret both towards the state and towards the public. We elaborate shortly on this case in Section 4.1.4.

[3] [Jarvenpaa et al., 2000] found that the reputation and the perceived size have a significant impact on the trust in internet stores

2.2 2-party service architectures

2.2.1 Basic idea

The data holder uses a service that is offered online (*web-based service*) that requires him to send input data to the service provider. The result of the service is returned to the data holder who is the service user at the same time (cf. Figure 2-2). A simple example is the query for specific share values. The name of the share (e.g. 'MERQ' for Mercury Interactive Group) is the input datum *D*, the result of the service request *S(D)* is ($42.46 24-Mar-04 3:58pm). Besides this basic kind of database query, there are more complex services such as wage accounting or online health checks. This case is particularly characterized by the absence of a third party, i.e. the data holder is simultaneously the service user and only has to protect his confidential data from the service provider. We assume for this case that the privacy policy of the service provider explicitly rules out the forwarding of customer / user information to a third party.

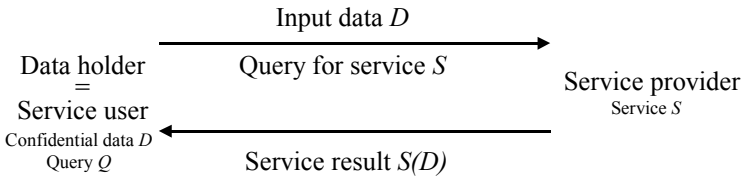


Figure 2-2: Confidential data flow in 2-party services

Note that in this case too, both input datum *D* and service result *S(D)* have to be protected from an untrusted service provider. Yet we will show in Section 3.3 that a transformation of *D* is sufficient to protect both *D* and *S(D)*.

2.2.2 Instances in real-world information systems

Although there are many application areas for the 2-party case, we will motivate our work with two important instances.

The *single-user web-based service* refers to the well-known case of a single person using popular web-based services e.g. when searching for information (e.g. at www.google.com), buying digital goods (e.g. at www.amazon.com) or managing financial assets (e.g. at www.citibank.com). A concerned user who is very hesitant with sharing confidential information with the service provider may ask the following questions.

- Can we look for information without letting the search provider know exactly what we are looking for?

- Can we buy a digital good without letting the online store know which music file or e-book we are interested in?
- Can we use an online portfolio service without letting the financial service provider know the total amount of our assets?

Although at least some of these tasks sound infeasible we will show in Chapter 3 that for a selected range of applications, we can indeed obtain a desired service result without sharing confidential input information with the service provider.

Opposed to the single-user web-based service, the *outsourced web-based service* refers to an *application service provider* (*ASP*) who offers "software as a service" usually to an entire company. An ASP "deploys, hosts and manages access to a packaged application for its customers from a centrally managed facility" (see [IDC, 1999] for a definition). Contracting an ASP promises its customers to reduce capital investment, to make IT costs more transparent, to facilitate the focus on core competencies and to provide faster access to high-end software applications [Tamm, 2003]. However, this also means that an ASP always hosts the (potentially confidential) business data of the customer with the corresponding implications for data security and privacy. An extensive survey by [Carter, 2000] shows that this innovative kind of software provision is severely inhibited by privacy concerns of ASP customers. We elaborate on these concerns in Section 3.2 and show how this conflict can be resolved for some particular applications.

2.2.3 Related work

Database services and arithmetic operations are core components of web-based services. Several approaches have been created to address related privacy problems in *database service provider architectures*.

2.2.3.1 Private Information Retrieval

One promising research direction is *Private Information Retrieval (PIR)* which was first presented by [Chor, et al., 1995]. It allows clients to query a database server, revealing neither the query nor the result of the query to the server. The model is simpler than that of traditional relational databases [Codd, 1970] because the query consists of an array index and the answer is the contents of the indexed array field. Yet it is powerful enough to implement many different applications such as file systems and dictionaries. If there is only one database server, the proven most efficient algorithm that preserves the secrecy of the query is for the user to download the entire database for each query. There are more efficient algorithms for several servers that do not cooperate in an attack. More recently, a practical method assuming a trusted physical device (a *secure coprocessor*,

see [Smith and Weingart, 1999]) in the server host has been proposed by [Asonov and Freytag, 2002]. This approach is almost optimal in resource overhead, and the assumption that the coprocessor on the server site is not compromised is arguably weaker than the assumption that several servers do not cooperate in an attack. PIR-algorithms can be used as a building block in privacy-preserving database outsourcing methods, see [Fischmann and Günther, 2003].

2.2.3.2 Partitioning and encryption

[Hacigumus, et al., 2002a; Hacigumus, et al., 2002b] present an approach that allows relational database operations on encrypted data. Before encryption, the data is aggregated to partitions on the customer side to decrease the amount of information that the service provider receives. The consequence is that the client needs to do some post-processing, as the provider can compute only an approximation of the result which can also be error-prone. Unfortunately, it is only efficient on a subset of relational algebra. For instance, each range query condition of the form $A < x$ needs to be transformed into a disjunction of conditions matching all concrete values smaller than x before being encrypted. Also, as [Fischmann and Günther, 2003] show, even with aggregation this scheme is not very secure.

[Damiani, et al., 2003] take a different route along the same line of reasoning. Instead of aggregating the data, each plaintext attribute is properly encrypted to a unique ciphertext, and a method is proposed to compute exposure coefficients that tell the data holder how much information he is giving away. Furthermore, it is explained how *B-trees* [Bayer and McCreight, 1972] can be encrypted to allow for more efficient range queries on encrypted tables. However, even encrypting the B-trees for retrieving ranges of records only helps improve performance with respect to the naive approach, but information on the attribute in question is still leaked. Each time all records are retrieved that satisfy $A < x$, the service provider learns a set of ciphertexts that represent values of A that are smaller than x.

[Song, et al., 2000] have proposed a family of schemes for encrypting a text corpus such that it can be searched without decryption. These methods are efficient and proven secure, and certainly an interesting building block for privacy-preserving application distribution.

As essential theoretical foundations we should mention *secure multi-party computation* [Goldreich, 1998], a generalization of privacy-preserving data mining and *oblivious transfer* [Naor and Pinkas, 2001], a more rigid category of protocols related to private information retrieval, although neither is the subject of this work.

2.2.3.3 Our contribution

We can see that the privacy-preserving use of arithmetic operations and database services have been elaborated in more or less disjoint research fields. However, we believe that both arithmetic and database operations are core components for almost every web-based service and should be analyzed and developed jointly.

In Chapter 3 we present a comprehensive analysis of what kind of services are feasible given the security requirements of the data holders. We explore how database and arithmetic operations can be usefully combined to offer securely outsourced services. Obviously, the extent of services that can be conducted on encrypted data is limited and not plentiful enough for arbitrary use. Our aim is to explore the trade-off between "not secure enough" and "not useful enough".

To motivate our work, we show how the confidential data of an ASP customer can be compromised. We propose a service architecture that hides plain data from the service provider and we carry out sample services within this framework. We evaluate our framework with regard to time and memory requirements and discuss practical implementation issues.

2.3 3-party service architectures

2.3.1 Basic idea

In a 3-party service architecture, the service user is not necessarily the data holder. This is the case in the scenario we sketched in the introduction, where a regional health initiative (the service provider) collects and analyzes data from patients (the data holders) to distribute results to medical researchers or, of course, to the patients (the service users). Figure 2-3 displays this.

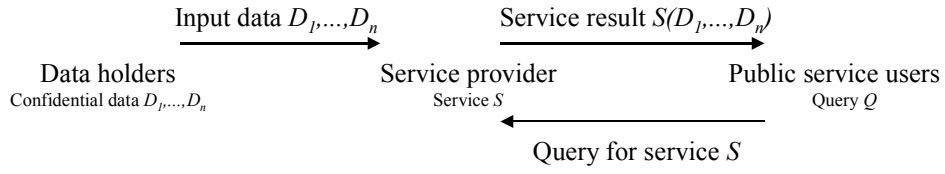


Figure 2-3: Confidential data flow in 3-party services

The main threat in this scenario is that from the published report (i.e. the service result), confidential information about individuals can be inferred. The problem of obtaining the confidential datum *D* from the publicly available service result *S(D)* is the *inference problem* well-known in the *statistical disclosure control (SDC)* literature (for comprehensive surveys on SDC, see [Adam and Wortman, 1989; Shoshani, 1982]).

2.3.2 Instances in real-world information systems

We will now give two examples for the 3-party service case.

Census bureaus such as the U.S. Bureau of Census (www.census.gov) collect data and provide statistics to the public in order to characterize regions socially and economically. The main threat for the data holders is the risk of being re-identified in the published statistic, thus divulging confidential information such as income or debt.

Regional health initiatives such as the Pittsburgh Regional Healthcare Initiative (PRHI, www.prhi.org) collect, analyze and disseminate chronic disease data to patients and researchers. Data holders include pharmacies, physicians, health maintenance organizations (*HMOs*), laboratories and patients. Besides the re-identification threat for patients, the other parties also fear a breach of their privacy. An HMO for instance may fear that internal data ends up in the hands of a competitor and is (mis-)used in marketing campaigns. We will elaborate on this case extensively in Chapter 4.

2.3.3 Related work

2.3.3.1 Data integration

Data integration deals with the technical integration of heterogeneous data sources. It is necessary for instance, when different HMOs provide data about the same diagnostic test in different formats. [Wiederhold, 1993] introduced the notion of a *mediator* between data holders and data providers to resolve semantic conflicts, and later added a security component to his basic model [Wiederhold, et al., 1996]. [Rezgui, et al., 2002] proposed a privacy mediator based on the screening of external database queries for sensitive attributes and their eventual removal from processed queries. Complementary to the mediator approach is the *data warehouse* approach [Chaudhuri and Dayal, 1997; Inmon, 1996], where data from the different sources is extracted, transformed and then loaded into a read-only database. Queries are then no longer run on the multiple databases via the mediator but directly on the data warehouse database. For reasons of simplicity, we will call the intermediary system "mediator" from now on because it best incorporates the idea of negotiating between data holders and service users.

2.3.3.2 Statistical disclosure control

Statistical disclosure control (SDC) is concerned with providing access to high-quality statistics for business / policy purposes (i.e., the service *S*) while at the same time protecting the confidentiality of the individual data providers (i.e. the data holders, for instance survey or Census respondents). SDC distinguishes two principle approaches,

query restriction and *data perturbation*. The query restriction family includes the following.

Query set size control [Fellegi, 1972] works by setting lower and upper bounds for the size of the query answer set based on the properties of the database and on the preferences fixed by the database administrator. If the number of returned records does not lie within these bounds, the information request would have to be rejected and the query answer is denied. As queries that are issued sequentially by one user often have a large number of entities in common, an improvement is the restriction of these entities to a maximum number, see [Dobkin, et al., 1979]. Although popular, this method is not robust enough as a stand-alone solution, see [Denning, 1982].

Auditing involves keeping up-to-date logs of all queries made by each user and constantly checking for possible disclosures whenever a new query is issued. One major drawback of this method is that it requires huge amounts of storage and CPU time to keep these logs updated. A well-known implementation of such an audit system is *Audit Expert* by [Chin and Özsoyoglu, 1982]. It uses binary matrices to indicate whether or not a record was involved in a query.

Cell suppression [Cox, 1980] is an important method for categorical databases when information is published in tabular form. Census Bureaus often make use of tabular data and publish counts of individuals based on different categories. One of the main privacy objectives is to avoid answers of a small size. For example, if a snooper knows somebody's residence, age and employer, he can issue a query for (`ZIP=10178, Age= 57, Employer= 'ABC'`). If the answer is one entity, the snooper could go on and query for (`ZIP= 10178, Age= 57, Employer= 'ABC', Diagnosis= 'Depression'`). If the answer is one again, the database is compromised and the person with the diagnosis identified. The cells must be suppressed. A common criterion to decide whether or not to suppress a cell is the *N-k rule* where a cell is suppressed if the top *N* respondents contribute at least k% of the cell total. *N* and *k* are parameters that are fixed by the database administrator, i.e. the Census Bureau. In the exemplary case of *N*= 2 and *k*= 10%, a cell which indicates aggregated income ($10M) of 100 individuals would have to be suppressed if the top two earners' aggregate income exceeded $1M.

In the query restriction approach, either exact data is delivered from the original database or the query is denied. An alternative is to perturb the original values such that confidential, individual data become useless for a snooper while the statistical properties of the attribute are preserved. The manipulated data is stored in a second database and is then freely accessible for the users.

Data swapping [Denning, 1982] is the process of exchanging attribute values (like income)

within a list of data holders such that no assignments from individuals to their income can be made anymore. However, the arithmetic average and the standard deviation of the attribute income stay the same.

Noise addition for numerical attributes [Traub, et al., 1984] means adding a disturbing term to each value: $Y_k = X_k + e_k$, where X_k is the original value and e_k adheres to a given probability distribution with mean zero. As for every value X_k, the perturbation e_k is fixed; therefore conducting multiple queries does not refine the snooper's search for confidential single values.

A hybrid approach are *random-sample queries* [Denning, 1980] where a sample is drawn from the query set in such a way that each entity of the complete set is included in the sample with probability *P*. If, for example, the sample of a COUNT query has *n* entities, then the size of the not perturbed query set can be estimated as *n/P*. If *P* is large, there should be a set-size restriction to avoid small query sets where all entities are included.

Another prominent approach to rule out re-identification in public databases is *k-anonymity* [Sweeney, 2001; Sweeney, 2002a; Sweeney, 2002b]. This is an anonymization procedure after which each individual cannot be distinguished from another *(k-1)* individuals in the database. Thus every set of attribute values appears at least *k* times. As *k* increases, so too does the anonymity in the database.

2.3.3.3 Privacy-preserving data mining

Data Mining, the science of efficiently discovering valuable, non-obvious information from large databases, may well be misused to intrude upon the privacy of organizations and individuals [Clifton and Marks, 1996]. One research direction is the use of cryptographic protocols to calculate aggregates from several contributors without divulging individual information [Canny, 2002a; Canny, 2002b; Clifton, 2001; Lindell and Pinkas, 2000; Vaidya and Clifton, 2002].

[Agrawal and Srikant, 2000] present a new method for privacy-preserving data mining. Based on a database of perturbed values, they are able to reconstruct the original value distribution. Confidential information of individuals is not compromised.

2.3.3.4 Our contribution

We propose a new class of privacy mediators that go beyond the state of the art with regard to the privacy protection methods applied to the final service result *S(D)*. We propose extending the common query-rewriting approach by developing and integrating new methods of disclosure control. Our proposals for preventing inferences are not limited

to mediator-based approaches and are applicable in the context of data warehousing and statistical databases as well, see Section 4.1.3.

To motivate our work, we use a real-world example that shows how a snooping HMO is able to determine narrow bounds on confidential information of its competitors by analyzing the aggregate data published by the mediator. We propose a specific "audit and aggregate" methodology that helps detect and prevent this specific kind of disclosure. Furthermore, we evaluate our method within a framework that measures the trade-off between decreasing risk of privacy breaches on behalf of the data provider and the loss of information for the legitimate service user.

2.4 A classification of typical services

2.4.1 Reactive vs. non-reactive data provision

After distinguishing between the two-party and the three-party case in the preceding sections, we will now introduce another dimension that is important in the privacy context. Users of web-based services can provide data either *reactively* to the service provider or *non-reactively / passively* (see e.g. [Boyens, et al., 2002]).

Reactive data provision means that data holders explicitly provide their personal data (e.g. for their health data in an online health check). They actively fill in web forms or web questionnaires and submit the information to the service provider.

Non-reactive or passive data provision means that data holders do not explicitly provide personal data but generate it automatically e.g. through their behavior. The best-known case for this is the tracking of web-shop customers with the help of cookies, a file stored on the user's hard disk that identifies the customer each time he visits a web site. For a more detailed description of cookies and their role in privacy protection see [EPIC, 2004b] or www.cookiecentral.com.

For the case of Internet services, [Teltzrow and Kobsa, 2004] distinguish between "user data" and "usage data", where user data refers to demographic data, user skills and knowledge, user interests and plans whereas usage data refers link selections, viewing behavior and purchase actions. Usually, online service users are much less aware of their passive data provision because it is often triggered by their web browser settings. For an overview of reactive and passive data provision, see Table 2-2.

		DATA PROVISION	
		Reactive	Passive
SERVICE	2-party	Online health check Financial portfolio service ASP / ERP services (***Web forms***)	Personalization services for shopping and travel (***Cookies***)
	3-party	Census (***Questionnaires***)	Chronic disease reports (***Primary care data***)

Table 2-2: Data provision in different services (means of data provision in parenthesis)

Allowing the employment of cookies does not only yield privacy compromises. It also yields benefits in the form of personalized services. These services can include customized finance pages or news collections, customized recommendations or advertisements based on past purchase behavior, customized pricing, express transactions or tailored email alerts. In online book stores for instance, these personalized services include recommendations to recently published books of interest. The recommendations are based on clicking behavior and on preceding book or CD purchases. Unfortunately, many online stores do not offer an option to (de-)activate the tracking of the click and purchase history and thereby prevent the user from easily trading off his own privacy concerns with the potential benefit from an extended service [Kobsa, 2001].

2.4.2 Sample services

In Table 2-3 and Table 2-4 we give a short and, of course, incomplete list of examples of services with their respective fit into the dimensions number of parties and reactive vs. passive data provision. For each service, we name examples of sensitive data that is typically required as input data, and we name some major threats that these data may be subject to.

2-party services				
SERVICE	SAMPLE SERVICE PROVIDER	DATA PROVISION	SENSITIVE DATA d_i	MAJOR THREATS / CRITICALITY
Health check	www. skolamed.de	reactive	age weight	Forwarding of personal health information to 3rd

			blood pressure cholesterol	parties
Online store	www. amazon.com	passive	hobbies interests shopping behavior	"Profiling" of customers, tracking via cookies
ASP service Wage accounting	÷	reactive	birth date income illnesses absence overtime	Disclosure of personal / corporate secrets (product launch dates)

Table 2-3: Typical services in 2-party architectures

3 -party services				
SERVICE	SAMPLE SERVICE PROVIDER	DATA PROVISION	SENSITIVE DATA d_i	MAJOR THREATS / CRITICALITY
Census	www. census.gov/	reactive	name address age profession religion	Re-identification of individuals, disclosure of e.g. income
Online voting	÷	reactive	election vote	Disclosure of the confidential vote either to the state or to fellow citizens
Health reports	www.phc4.org	passive	address diagnosis blood test results, eye exam results, lipid profiles	Re-identification of individuals / disclosure of e.g. confidential test values

Table 2-4: Typical services in 3-party architectures

2.5 What this thesis is not about

There are several areas of research that are outside the scope of this thesis. The most important one is technical data security. We assume that adequate measures for communication confidentiality (such as the Secure Socket Layer *SSL* [Netscape, 1996]) and access control (such as Kerberos [Neumann and Ts'o, 1994] or X.509 [ITU, 2000]) are in place. For a detailed discussion of cryptography and network security, see [Schneier, 1996; Stallings, 1999].

Another important area is the extension of privacy legislation. For purposes of this work we assume the validity of the current privacy laws and recommendations in place, in particular [HIPAA, 1996; USPA, 1974] for the USA and [EU, 1995; EU, 2002] for the European Union.

A very important area of investigation of an individual's attitude towards privacy [Ackerman, et al., 1999] and the behavior that is derived from this attitude (which often differs significantly from the attitude declared beforehand, see [Spiekermann, et al., 2001]). Our work focuses on technical measures that are not affected by individual privacy behavior.

3 Protecting sensitive information in data for web-based services

Privacy is the right to be let alone - the most comprehensive of rights, and the right most valued by civilized men.

(Louis Brandeis, US supreme court justice)

3.1 Motivation

As already discussed in Section 2.1.1, a well-known business model for the 2-party web-based service is the *Application Service Provider (ASP)*. An ASP "deploys, hosts and manages access to a packaged application to multiple parties from a centrally managed facility. The applications are delivered over networks on a subscription basis. This delivery model speeds implementation, minimizes the expenses and risks incurred across the application life cycle, and overcomes the chronic shortage of qualified technical personnel available in-house" [IDC, 1999]. For these reasons, the ASP model has significant business impact and was forecasted annual growth rates between 75% and 89% [Mizoras, et al., 2001; Terdimann, et al., 2000].

Speaking in terms of the definition in Section 2.1, the ASP is the service provider and the ASP customer is the data holder (who is, in this case, the service user at the same time). We will refer to this business model throughout the rest of the chapter.

3.2 Privacy concerns for users of web-based services

Yet one reason that inhibits the wide spread use of this new kind of services is the question of data ownership and confidentiality. Classical Enterprise Resource Planning (ERP) installations that were based on the purchase and local installation of software and hardware ensured that confidential data did not leave the customer company's premises. In the ASP model this is no longer the case. Using software services over the Internet requires the customer to transfer potentially sensitive business data to the service provider which may include internal financial figures, product launch schedules and the like.

This obviously raises security concerns on behalf of the customers who urge the ASP to

use firewalls, dedicated servers and encryption of the communication channel to protect their most confidential business data. These protective measures may, however, not suffice as the data is still required in unencrypted form for the ASP to process and to deliver the actual service results. In Table 3-1, we describe four different kinds of attacks. With exception of the first threat, an external attack against the customer's database, prevention along the conventional lines is very difficult.

#	THREAT	DESCRIPTION	RISK INDICATORS
1	External attack against the customer's database	External attacks directed at the service provider's database are still possible, and the risk is hard to estimate	Audit of the ASP's system security
2	Malicious ASP staff	Malicious staff on the provider's side (bribed or disgruntled employees etc.) may want to cause harm to their company and its customers	Fluctuation rate of ASP employees
3	Incompetent ASP staff "Social Engineering"	Incompetent staff on the provider's side may unintendedly grant data access to unauthorized parties	Staff / workload ratio
4	Bankruptcy of the ASP or change of ownership	Bankruptcy or acquisition of the ASP leads to the transfer of the customer's business data, in the worst case to a direct competitor of the customer	Financial and competitive position of the ASP

Table 3-1: Threats to confidential business data of ASP customers

External attacks are usually accounted for with adequate cryptographic and organizational measures but cannot be completely ruled out. The damage caused by disgruntled or malicious ASP staff is even harder to predict and prevent. The CSI/FBI Computer and Crime Survey [CSI, 2003] shows that disgruntled employees are the most likely source of attacks, even more likely than independent hackers or competitors (86% vs. 74% / 53%, resp.). The risk of an attack by disgruntled employees is hard to measure. A possible indicator might be the number of employees who leave the ASP company per year (fluctuation rate).

Although applications like online banking and online book stores have become ubiquitous, people working in sensitive areas in IT companies are still vulnerable to trivial attacks

called *social engineering* [Rusch, 2004]. One example of social engineering is walking into an office, telling an unsuspecting person that you need to fix a problem with the intranet and therefore need her password. The probability of getting the password is higher than one might expect.

Finally, the potential consequences of bankruptcy and/or change of ownership of the provider may be serious. Online retailer amazon.com's privacy statement clearly states that customer data is sold in the case of a change of ownership.

> "…we might sell or buy stores, subsidiaries, or business units. In such transactions, customer information generally is one of the transferred business assets... Also, in the unlikely event that Amazon.com, Inc., or substantially all of its assets are acquired, customer information will of course be one of the transferred assets…" [Amazon.com, 2004]

Technically, no corporate purchase can change the validity of a privacy policy that the former company has agreed to, but even if the new data owner is legally bound to a privacy policy, enforcing this in an international law suit is often infeasible. In the worst case, a direct competitor of one of the provider's customers might end up owning all the outsourced business data. However unlikely, this scenario has considerable potential to scare customers.

3.3 A privacy-preserving architecture

In this section, we present a service architecture that allows for processing data with a very high level of privacy protection. Sensitive data is not only withheld with respect to non-trusted third parties, but also to the service provider itself. The service provider will not dispose of *any* unencrypted customer data at *any* time. Contrary to the concept of [Asonov and Freytag, 2002], no hardware equipment is involved. Our approach requires the service provider to work directly with encrypted data.

Following the approach of *public key infrastructure* first proposed by [Rivest, et al., 1978a; Rivest, et al., 1978b], the basic idea is to transform the sensitive data with the help of a secret key only known to the customer. The service provider uses the corresponding public key in order to process the encrypted data. Without the private key, the service provider cannot see any sensitive information in plaintext (as is intended by the customer). Without the public key, it cannot even compute the data.

From an infrastructure point of view, the architecture requires the following actions.

- The creation of a private key and its safe-keeping.
- The creation of a public key and distribution of it to the service provider.
- Equipment of customer software with the transformation algorithm.

- The adaptation of the service provider's business logic such that encrypted data can be processed.

How these requirements are dealt with in practice is discussed in Section 3.8.3. The volume of infrastructure requirements implies that the approach is more suitable for Application Service Provider (ASP)-like solutions which allow at least for some customization than for fine-grained, standardized web services. We use the following terminology.

d_i, i=1..n	Sensitive input data from the data holder
p	Private key of the data holder
q	Public key (given to the service provider)
S: $(d_i, d_j) \rightarrow S(d_i, d_j)$	Operation / Service on plain customer input data
T_p: $d_i \rightarrow T_p(d_i) = t_i$	Encryption / Transformation function
S': $(t_i, t_j) \rightarrow S'(t_i, t_j)$	Operation / Service on encrypted customer data
T_p^{-1}: $t_i \rightarrow T_p^{-1}(t_i)$	Decryption / Retransformation function

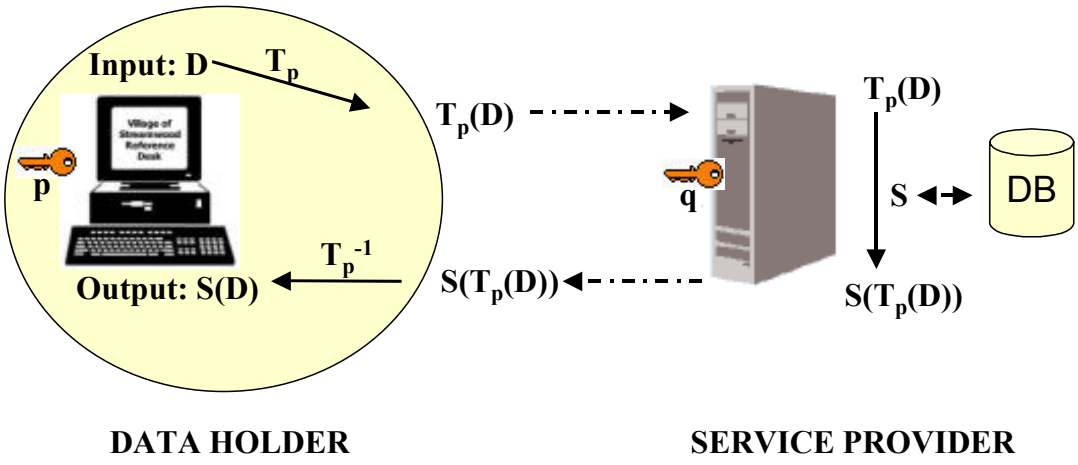


Figure 3-1: A sketch of the proposed service architecture

Figure 3-1 describes the general service procedure. The customer wants the service provider to perform some service S on the confidential data D she provides. After installing the key infrastructure, critical data is marked up as “sensitive” and the application running on the customer machine encrypts it using the provided transformation

scheme *T* and the private key *p*. The server, who only sees encrypted data, now uses the public key *q* to perform the requested service *S*. Once the server has performed its service, the encrypted pseudo-solution $S(T_p(D))$ is retransferred to the customer who applies a re-transformation (usually T_p^{-14v}) to obtain the desired result $S(D)$.

The whole procedure is summed up in Figure 3-2.

Procedure: SERVICE_PROVISION (D, S, T, p, q)

Input: Confidential data D. Service S. Transformation scheme T. Private key p. Public key q.

Output: A service result S(D)

Steps

//Install key infrastructure

Create two large secret primes p and p'

Compute q:= p p'

Let p be the private key

Let q be the public key

Keep p and transfer q to the service provider

//Encrypt confidential data

Encrypt D with T_p and get $T_p(D)$

Transmit $T_p(D)$ to the service provider

//Service provision

Let the service provider calculate $S(T_p(D))$

//Decrypt confidential result

Receive $S(T_p(D))$ from the service provider

Decrypt the encrypted result with T^{-1}

Get $T^{-1}(S(T_p(D))) = S(D)$

Figure 3-2: Steps of the proposed service architecture

3.4 Data transformation

The "transformation schemes" *T* that we referred to in the previous section are actually *encryption functions* in the cryptographic terminology. These functions map *plaintext*, the readable sensitive data, to *ciphertext*, its encoded counterpart. Cryptanalysts determine

[4] Note that the retransformation is not necessarily the inverse function T^{-1}. For reasons of simplicity, we work just with T^{-1} throughout the rest of the chapter.

the security of an encryption scheme in terms of its resistance against six attacks of increasing scale. [Stallings, 1999] and [Schneier, 1996] give an overview of the different attacks that we summarize in Table 3-2.

TYPE OF ATTACK	KNOWN TO ATTACKER	DIFFICULTY OF ATTACK
No encryption algorithm	- Ciphertext to be decoded	Most difficult ↑
Ciphertext only	- Ciphertext to be decoded - Encryption algorithm	
Known plaintext	- Ciphertext to be decoded - Encryption algorithm - One or more plaintext-ciphertext pairs formed with the secret key	
Chosen plaintext	- Ciphertext to be decoded - Encryption algorithm - Plaintext message chosen by attacker, together with its corresponding ciphertext generated with the secret key	
Chosen ciphertext	- Ciphertext to be decoded - Encryption algorithm - Purported ciphertext chosen by attacker, together with its corresponding decrypted plaintext generated with the secret key	
Chosen text	- Ciphertext to be decoded - Encryption algorithm - Plaintext message chosen by attacker, together with its corresponding ciphertext generated with the secret key - Purported ciphertext chosen by attacker, together with its corresponding decrypted plaintext generated with the secret key	Least difficult ↓

Table 3-2: Difficulty of cryptographic attacks

The architecture presented in this Chapter is based on a particular class of encryption functions, so-called *privacy homomorphisms (PHs)*. [Rivest, et al., 1978a] introduce them as "encryption functions that permit encrypted data to be worked with without preliminary decryption of the operands". We now define the homomorphic property of privacy homomorphism.

Definition (Homomorphic encryption function)

An encryption function T_p: $d_i \rightarrow T_p(d_i)$ is *homomorphic* with regard to a Service S iff $\forall d_i, d_j \in \text{dom}(T)$: $T^{-1}(S'(T(d_i), T(d_j)) = T^{-1}(T(S(d_1, d_2))) = S(d_1, d_2)$

Referring to Figure 3-3, this means that it does not matter whether you first perform the service on plaintext and then encrypt the data or whether you first encrypt the confidential data and perform the service on encrypted data. However, working with encrypted data imposes significant restrictions on the extent of feasible operations on the underlying data. There are two fundamental findings with important implications.

First, a secure encryption scheme can never preserve order. This means that for a given set of attribute values such as e.g. (3; 17; 31; 35; 42), no secure encryption scheme allows for determining the order ($T_p(3)$, T_p (17), T_p (31), T_p (35), T_p (42)) on the ciphertexts [Rivest, et al., 1978a]. An important conclusion is that attributes in encrypted databases cannot be sorted at all.

Second, the multiplication operation can be preserved at a higher level of security than addition. On the scale that measures the security of encryption schemes (see Table 3-2), the maximum level of security for addition-preserving encryption schemes is known-plaintext-resistance [Brickell and Yacobi, 1987]. This is less secure than the maximum level for multiplication-preserving ones, chosen-ciphertext resistance. The implication is that encryption schemes that preserve all basic arithmetic operations can reach at most the lower additive security level.

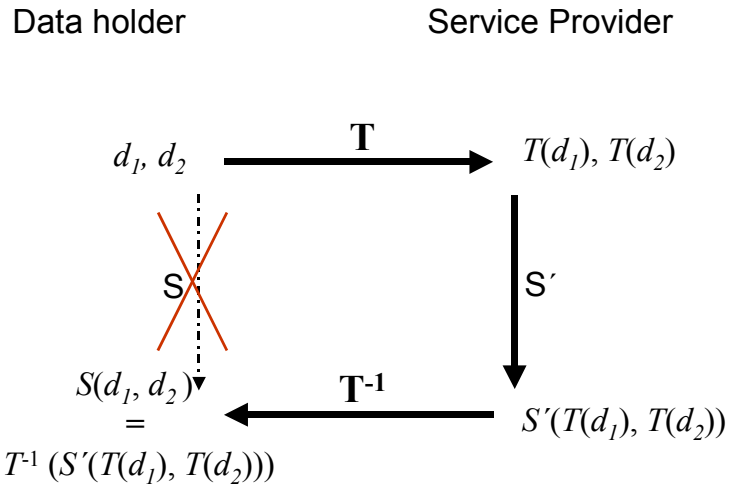


Figure 3-3: The basic idea of a privacy homomorphism

To employ PHs in service environments, the basic idea would be to transfer the encrypted confidential data $T(d_1)$ and $T(d_2)$ to the service provider instead of the plaintext pair d_1 and d_2. The provider then computes the pseudo-solution $S'(T(d_1), T(d_2))$ which is, due to T 's homomorphic property, equal to $T(S(d_1 \cdot d_2))$. Applying the inverse function T^{-1} then decrypts the pseudo-solution and yields the desired result $S(d_1,d_2)$. The three solid arrows in Figure 3-3 clarify this procedure.

Example: The sample PH that [Rivest, et al., 1978a] describe yields that the multiplicative product of two encrypted numbers is equal to the encryption of the corresponding plaintext product.

$$T(d_1)\cdot T(d_2)= T(d_1\cdot d_2)$$

Applying the inverse function gives

$$T^{-1}(T(d_1)\cdot T(d_2))= T^{-1}(T(d_1\cdot d_2))= d_1\cdot d_2$$

In this case, the provided service *S* is the multiplication of numbers.

If one considers “multiplication” as a simple kind of service, the encryption function *T* thus guarantees a very high level of privacy protection because the customer may use the service while revealing neither the factors nor the result to the service provider.

Whereas this multiplicative PH is a very secure one (chosen-ciphertext-resistant), performing addition on encrypted data turns out to be a more complicated issue. [Ahituv, et al., 1987] show that an additive PH may reach at most known-plaintext-resistance. [Brickell and Yacobi, 1987] are the first to present an *R*-additive PH that permits the addition of up to *R* numbers with ciphertext-only-resistance. Finally, [Domingo-Ferrer and Herrera-Joancomarti, 1999] present a PH allowing all field operations (addition, subtraction, multiplication and inverse multiplication) on an arbitrary number of ciphertexts. Though it is ciphertext-only resistant, it would still force the potential attacker to acquire plaintext information from the customer in order to be successful, which transfers at least some of the responsibility from the service provider to the customer (see [Boyens and Günther, 2002]). If a plaintext-ciphertext pair has been known to the attacker, it will be difficult for the customer to deny at least part of the responsibility for the break-in. A practical solution for this problem is to codify these responsibilities in the service contract. The customer would then be at least partially responsible for a potential break-in. See Table 3-3 for an overview of existing privacy homomorphisms.

PHs have been employed for very specific purposes such as multi-application smart-cards [Domingo-Ferrer, 1997], signature schemes [Johnson, et al., 2002] and electronic voting [Asonov, et al., 2001]. The field of research with most potential impact however is the field of securely outsourced database services, see [Damiani, et al., 2003; Hacigumus, et al., 2002a; Hacigumus, et al., 2002b; Ozsoyoglu, et al., 2003] and our discussion in Section 2.2.3.2.

AUTHORS	SERVICE S	SECURE AGAINST	REMARKS
[Rivest, et al., 1978a]	$S(d_1, d_2) = d_1 \times d_2$	Chosen-ciphertext attack	Based on RSA Preserves equality
[Brickell and Yacobi, 1987]	$S(d_1, ..., d_R) = \sum_{i=1}^{R} d_i$	Ciphertext-only attack	First R-additive scheme
[Domingo-Ferrer, 1996]	$S(d_1, d_2) = d_1 + d_2$ $S(d_1, d_2) = d_1 \times d_2$	Known-plaintext attack	
[Domingo-Ferrer and Herrera-Joancomarti, 1999]	$S(d_1, d_2) = d_1 + d_2$ $S(d_1, d_2) = d_1 - d_2$ $S(d_1, d_2) = d_1 \times d_2$ $S(d_1, d_2) = d_1 \div d_2$	Ciphertext-only attack	Supports all field operations

Table 3-3: Overview of existing privacy homomorphisms

We will now describe the PH in use with the proposed architecture.

3.5 The deployed privacy homomorphism

3.5.1 Encryption

The PH we base our architecture on is adapted slightly from the scheme proposed by [Domingo-Ferrer and Herrera-Joancomarti, 1999]. We apply the procedure depicted in Figure 3-4 to encrypt plaintext. Note that this scheme differs from the PH proposed by [Domingo-Ferrer and Herrera-Joancomarti, 1999] in the sense that $a \in \mathbf{Z}_p$ is *not* chosen arbitrarily but as a fixed and secret prime. As modular equations $a \cdot x = d \pmod{p}$ have unique solutions for $a, x, d \in \mathbf{Z}_p$, unique plaintext identifiers have the same ciphertext correspondents. This would not have been the case if *a* had been chosen arbitrarily. This feature is important since, for instance, primary keys for a database can now be addressed by a unique ciphertext. The check for equality allows for picking single records out of the encrypted database, thus permitting the updating, deleting and retrieving of records that already exist in the database.

Procedure: ENCRYPT (d, p, q, a))

Input: Confidential plaintext data d. Private key p. Public key q. A secret number a.

Output: An encrypted confidential datum $T_p(d)$

Steps

//Install key infrastructure

Create two large secret primes p and p'

Compute q:= p p'

Let p be the private key

Let q be the public key

Keep p and transfer q to the service provider

//Encrypt confidential data

Pick a secret number $a \in \mathbf{Z_p}$

Solve the modular equation $a \cdot x \equiv d \pmod{p}$ for $x \in \mathbf{Z_p}$

$T_p(d) := a \cdot x \pmod{q}$, $T_p(d) \in \mathbf{Z_p}$

Figure 3-4: Encryption procedure

3.5.2 Decryption

The decryption works in a similar manner. The difference consists of the fact that *A* can be chosen arbitrarily, as the transformation scheme guarantees the plaintext originally provided as the result.

Procedure: DECRYPT (t, p, q))

Input: Encrypted data $t=T_p(d)$. Private key p. Public key q.

Output: A decrypted confidential plaintext $T^{-1}(t)= d$

Steps

//Decrypt confidential data

Pick $A \in \mathbf{Z_p}$ arbitrarily

Solve the modular equation $A \cdot y \equiv t \pmod{q}$ for $y \in \mathbf{Z_q}$

$T_p^{-1}(t) := A \cdot y \pmod{q}$, $T_p^{-1}(t) \in \mathbf{Z_p}$

Figure 3-5: Decryption procedure

Note that modular equations of the type $a \cdot x \equiv d \pmod{p}$ for $x \in \mathbf{Z_p}$ are solvable if *p* is a prime and that the solution is unambiguous [Fieger, 1996].

3.5.3 A simple example

For the simple service of "multiplication", we will now give an example. We choose $d_1 = 3$ and $d_2 = 5$ and let the service provider calculate the service result $S(d_1, d_2) = d_1 \cdot d_2$.

Example: $p = 17$

p'= 31

a= 13, $a \in$ {2, 3, 4, …, 16}

q= 17·31= 527

d_1= 3 ($\in \mathbf{Z_{17}}$)

d_2= 5 ($\in \mathbf{Z_{17}}$)

//Encrypt confidential data

→ Solve $a \cdot x$= 13·x= 3 (mod 17) → x= 12; $a \cdot x$= 156 (mod 527)= $T_{17}(3) \in \mathbf{Z_{527}}$

→ Solve $a \cdot x$= 13·x= 5 (mod 17) → x= 3; $a \cdot x$= 39 (mod 527)= $T_{17}(5) \in \mathbf{Z_{527}}$

//Service provision

$T_{17}(3) *_{(\text{mod } 527)} T_{17}(5)$ = 156·139= 6084 (mod 527)= 287

//Decrypt confidential result

Pick A arbitrarily $\in \mathbf{Z_q}$: A= 412

Solve $A \cdot Y$= 287 (mod 527), $A \cdot Y$= 25056

$T_{17}^{-1}(T_{17}(3) *_{(\text{mod } 527)} T_{17}(5))$= 25056 (mod 17)= 15

You can see that neither the encrypted input data nor the encrypted result is of any use or meaning to the service provider. However, the result is still valid for service user.

3.6 Enabled services: Which services can be performed

Now that the basic service idea and the corresponding transformation scheme are introduced, we will discuss *which* actual services the service provider is able to carry out on the modified information he possesses. Naturally, encrypted data cannot be processed with the same range of operations as unencrypted data. We will distinguish between two different elementary service types.

The first elementary service type concerns basic database queries, such as retrievals and updates. We will analyze the basic relational operators concerning their suitability for handling encrypted records and give examples in Section 3.6.1.

The second elementary service type consists of the basic arithmetic operations, addition and subtraction, multiplication and division. We will show to what extent and on which kind of plaintext data these operations can be applied in Section 3.6.2.

3.6.1 Database services

Here we introduce a service for our running ASP example. The customer is a company who wants to outsource its Human Resource (HR) Management System, i.e. it wants the ASP to store and process employee information such as loans, overtime, etc. For that purpose, it transfers information about its employees and about the monthly wage accounts in the following two tables.

- `employee (employee_no, name, year_of_birth, department);`
- `monthly_account (employee_no, month, absence, overtime, payment);`

The `employee` table contains general information about the staff such as employee number, name, year of birth and department. A typical data record would contain the following.

```
(432321, 'Schmidt', 1963, 'Finance')
```

The `monthly_account` table in contrast yields information about the monthly payment account as absent hours, overtime hours and payment.

```
(432321, 'AUG 2002', 12, 23, 3247)
```

We will now explain if and how standard Structured Query Language (SQL) queries can be mapped such that the encrypted database can be accessed.

Selection

`("SELECT name, year_of_birth FROM employee` **`WHERE (department='Finance')`**`")`

The value to retrieve is simply encrypted in the query

"...`WHERE department=`T_p`(ascii('Finance'))`", where *ascii('Finance')* would be the corresponding ASCII coding. The exact and complete value must be specified, as the transformation scheme does not allow for "partial encryption". Therefore, working with wildcards ("...`WHERE (department LIKE 'F%')`") is not possible.

Projection

("**`SELECT name, year_of_birth`** `FROM employee WHERE (department='Finance')`")

Projection is possible without restrictions, as usually all the attribute names must be specified with their exact and complete names. Furthermore, it is up to the customer to decide whether he should just encrypt the values or encrypt the attribute names, too. In the latter case, the query would start with:

("`SELECT` T_p`(name),` T_p`(year_of_birth)…`")

Join

(“SELECT payment FROM employee e, monthly_account m WHERE **(e.employee_no = m.employee_no)**”)

The Join command for data from different tables works well as long as the matching is done with complete attributes (no wildcards). The privacy homomorphism guarantees that identical unencrypted values will have the same ciphertext correspondent. For example, T_p(employee_no) will be the same in table employee as in the table monthly_account.

Sorting

(“SELECT name, year_of_birth FROM employee **SORT BY year_of_birth)**”)

The ability to sort presumes the existence of a total order over the encrypted data. However, [Rivest, et al., 1978a] show that PHs that preserve total order in spite of the transformation cannot even be ciphertext-only resistant. Therefore, “SORT BY” cannot be conducted *at all* over encrypted data. An approach concerning how to facilitate this with some involvement of the customers' machines was recently proposed by [Hacigumus, et al., 2002b].

In order to *modify* the encrypted database, additional operators are necessary for record insertion, deletion or updating. However, they all depend on the discussed query operators. Hence e.g. deletion is possible for specifically selected values, but not for wildcard values. As a result, all records whose name attribute is equal to “Miller” could be deleted, but not those with name attributes starting with “M%”, as discussed for the “Selection” operator.

Table 3-4 sums up these results

OPERATOR	FEASIBLE ON T_P(D)?	REMARKS
Selection	Partially	No wildcard selection possible
Projection	Yes	Attribute name not necessarily encrypted
Join	Partially	Only over exactly matching data
Sorting	No	Impossible on secure data

Table 3-4: Database query operators on encrypted data

3.6.2 Arithmetic operations

All arithmetic operations discussed are principally *modular* operations. Yet on the plaintext domain $\mathbf{Z_p}$, the very large prime p allows for the calculation of large sums and products

without creating remainder terms through division by *p*. Hence addition, subtraction and multiplication can normally be used as if the algebraic space was the regular algebraic ring (**Z**, +, *). Furthermore, as ($\mathbf{Z_p}$, $+_{\text{mod p}}$, $*_{\text{mod p}}$) is equivalent to an algebraic field, it allows for the computation of multiplicative inverses. All of these properties are transferred to the algebraic space ($\mathbf{Z_q}$, $+_{\text{mod q}}$, $*_{\text{mod q}}$) after applying the transformation scheme presented in Section 3.5. The basic difference between ($\mathbf{Z_p}$, $+_{\text{mod p}}$, $*_{\text{mod p}}$) and ($\mathbf{Z_q}$, $+_{\text{mod q}}$, $*_{\text{mod q}}$) lies in the fact that *every* unencrypted datum is converted into a cipher of almost the same bit length as *q*, i.e. up to 256 bits. That means that e.g. the addition of salaries, say of 3275$ and 4023$ turns from the addition of 12-bit-integers to the addition of its 256-bit-long encrypted correspondents.

In the following, we will discuss the four basic arithmetic field operations. Afterwards, we will indicate for which aggregate operations the algorithm fits best.

Addition $d_1 + d_2 := T_p^{-1}(T_p(d_1) +_{(\text{mod q})} T_p(d_2))$

The regular (non-modular) addition of the unencrypted data is mapped to the modular addition of the encrypted numbers. It works for all $d_1, d_2 \in \mathbf{Z_p}$, as long as $[d_1+d_2 < p]$, which is not a strong condition because *p* is large.

Subtraction $d_1 - d_2 := T_p^{-1}(T_p(d_1) -_{(\text{mod q})} T_p(d_2))$

As $\mathbf{Z_p}$ does not contain negative integers, this only works as long as $d_1 > d_2$. From $[(d_1 - d_2) > 0]$ and $[(d_1-d_2) < d_1 < p]$ then follows $[(d_1-d_2) \in \mathbf{Z_p}]$

Multiplication $d_1 * d_2 := T_p^{-1}(T_p(d_1) *_{(\text{mod q})} T_p(d_2))$

This works as regular (non-modular) multiplication as long as $[d_1*d_2 < p]$. This can actually turn out to be a strong condition if the number of factors is very high.

Inverse Multiplication $d_1 * d_2^{-1} := T_p^{-1}(T_p(d_1) *_{(\text{mod q})} T_p(d_2)^{-1})$

This only works as the common "division" as long as d_2 in fact divides d_1. If division leads to a remainder, one may still compute the multiplicative inverse of $T_p(d_2)$, but the decrypted product does not correspond to the a readable figure (as d_1 *DIV* d_2, the integer division, would). It should therefore only be used as the regular division when the property "d_2 divides d_1" can be ensured beforehand.

Table 3-5 sums up these findings.

OPERATION	FEASIBLE ON $T_P(D)$?	CONDITIONS
Addition	Yes	$d_1+d_2 < p$
Subtraction	Partially	$d_1-d_2 > 0$
Multiplication	Yes	$d_1*d_2 < p$
Division	Partially	$d_2 \mid d_1$

Table 3-5: Arithmetic operators on encrypted data

3.7 Practical services

In this section, we will discuss a few sample services based on the encrypted `employee` and `monthly_account` tables presented in the previous paragraph. We think that HR data is particularly appropriate for this purpose, for two reasons. First, sensitive data can be found in various forms such as regular wages and bonus payments, absent and overtime hours, and sometimes even church affiliation. Second, HR Management tools are often subject to outsourcing and therefore represent a suitable application field for the proposed architecture.

S_1: Mean monthly absent hours in specific departments

Formally, this figure is calculated as the average μ_i over the absent hours of the employees *e* in department *dep1*

$\mu_{dep1}=(\Sigma_{(e.department= dep1)}\ e.absence) / |\{e \mid e.department= dep1\}|$

In order to calculate the mean absent hours for the 'Finance' department in August, the following actions are required on the provider's part.

1) Retrieve the `absence` attribute of all employees in the finance department.

   ```
   SELECT absence AS department_absence FROM employee e, monthly_account
   m WHERE (e.employee_no = m.employee_no) AND (e.department =
   Tp('Finance'))
   ```

   Note that this query includes a join over encrypted data, namely the employee number in both tables.

2) Calculate the sum over `department_absence`.

   $sum_{'Finance'}= \Sigma_{(e.department= Tp('Finance'))}\ e.absence$

3) Return the encrypted $sum_{'Finance'}$ and the plain $record\_count_{'Finance'} = |\{e \mid e.department = 'Finance'\}|$ to the customer.

4) Finally, the customer decrypts the sum and divides it by the count to obtain the result μ_{dep1}.

$\mu_{'Finance'} = T_p^{-1}(sum_{'Finance'}) / record\_count_{'Finance'}$

Note that lacking the possibility of dividing the two numbers leads to at least some involvement of the customer. A good example for a service that does not need any kind of customer intervention is the multiplication of matrices, as only multiplication and addition is required.

S_2: <u>Standard deviation of payments among departments</u>

This metric measures the income disparities among different departments. We will use a service similar to S_1 to calculate μ_i^* , the mean incomes per department.

$\sigma_{all} = ((\Sigma_{(departments\ i)} |\mu_{all} - \mu_i^*|^2) / |\{ i \mid dep_i \text{ is department } \}|)^{1/2}$

with $\mu_{all} = (\Sigma_{(departments\ i)} \mu_i^*) / |\{ i \mid dep_i \text{ is department } \}|$

1) Compute the mean payments μ_i^* for all departments using a similar service to S_1.

2) Compute the average μ_{all} over all μ_i^* 's using S_1 again.

3) Compute the sum of the squared differences: $squared\_dev := \Sigma_{(departments\ i)} |\mu_{all} - \mu_i^*|^2$.

4) Return squared_dev and the number of departments department_count to the customer.

5) The customer decrypts the squared deviation sum, divides it by the department count and draws the square root.

Again, some customer involvement is required. However the major part of the calculation is done by the provider, which especially pays off if the underlying databases are large.

3.8 A prototypical implementation

3.8.1 Sketch of the implementation

In order to evaluate the proposed architecture, we implemented a prototype of the service architecture. Figure 3-6 displays a sketch of the implementation and the employment of different Java classes.

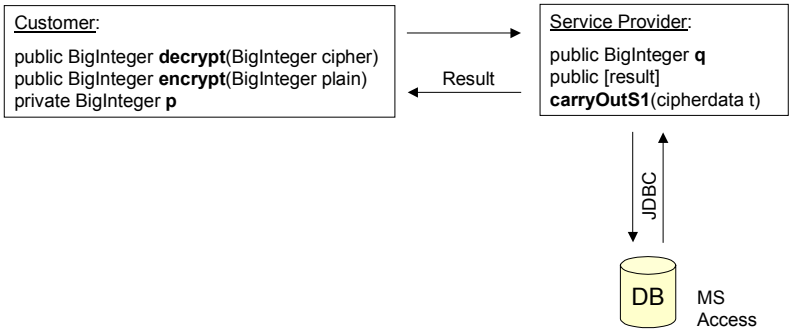


Figure 3-6: Sketch of the implementation

We chose Java as the programming language because it has convenient classes and methods to process large integers such as the secret and public primes as well as the encrypted data. The implementation was carried out with the technological components displayed in Figure 3-10.

COMPONENT	TECHNOLOGY	REFERENCE
CPU	x86 (700 Mhz)	h18000.www1.hp.com/products/quickspecs/10382_ca/10382_ca.html
RAM	192 MB	
Operating System	MS Windows 2000	http://www.microsoft.com/windows2000
Programming language	Java 1.4.2	java.sun.com/j2se/1.4.2
Database management system	MS Access 2000	office.microsoft.com/home/default.aspx

Table 3-6: Technological components of the implementation

3.8.2 Experiments

We created the employee and monthly_account tables with n=1000 data records. We first built them with unencrypted test data. Then we encrypted them using the proposed algorithm and a 32 bit, a 64 bit and a 128 bit key. As the focus is on the protection of sensitive data, we pay particular attention to the transformation of hours absent, overtime and salary (payment). The resulting tables have the following shape.

employee (T_p(employee_no), name, year_of_birth, department);

monthly_account (T_p(employee_no), month, T_p(absence), T_p(overtime), T_p(payment));

We first compared the service execution time of the service S_1 with regard to the size of the encryption key, see Figure 3-7.

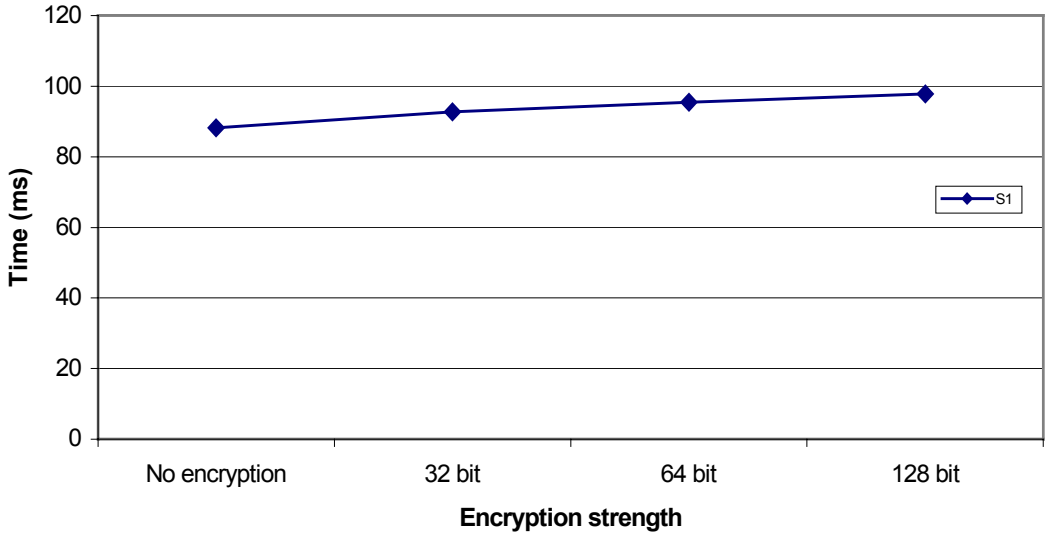


Figure 3-7: Service execution time with regard to encryption key length

The service performance time increases only slightly with the key size. Note that the service performance time does not include the creation or modification of the encrypted data in the customer's database but only the request for the mean monthly absent hours in the 'Finance' department. The time that is necessary to create `employee` and `monthly_account` as encrypted tables in the customer database with key sizes of different length is displayed in Figure 3-8.

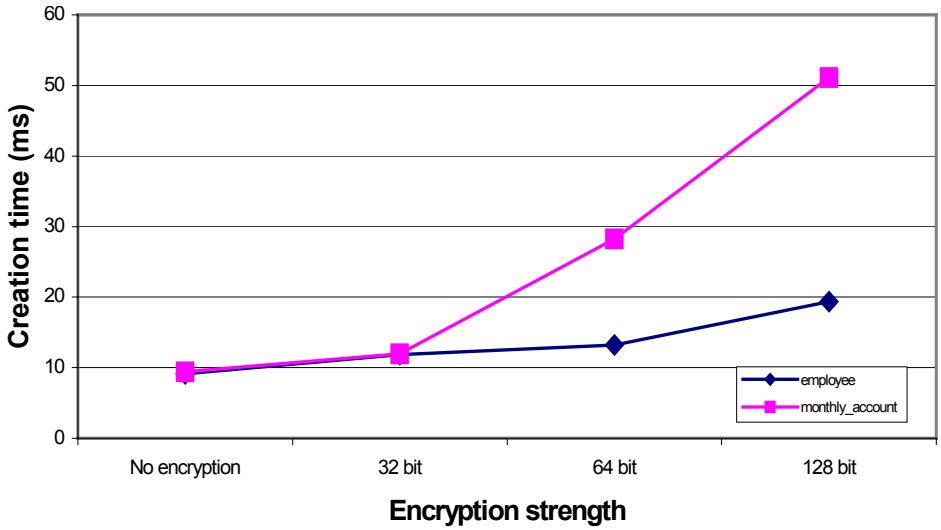


Figure 3-8: Table creation time with regard to encryption key length

We can see a significant increase in the creation time with growing key size. This is

particularly true for the `monthly_account` table. The reason for this is that `monthly_account` has more encrypted attributes (`employee_no`, `payment`, `absence`, `overtime`) than `employee` (just `employee_no`). Though differences were expected, the extent of the size gap is surprising.

Figure 3-9 shows another important dimension for database management systems, the size of the encrypted tables for different key lengths.

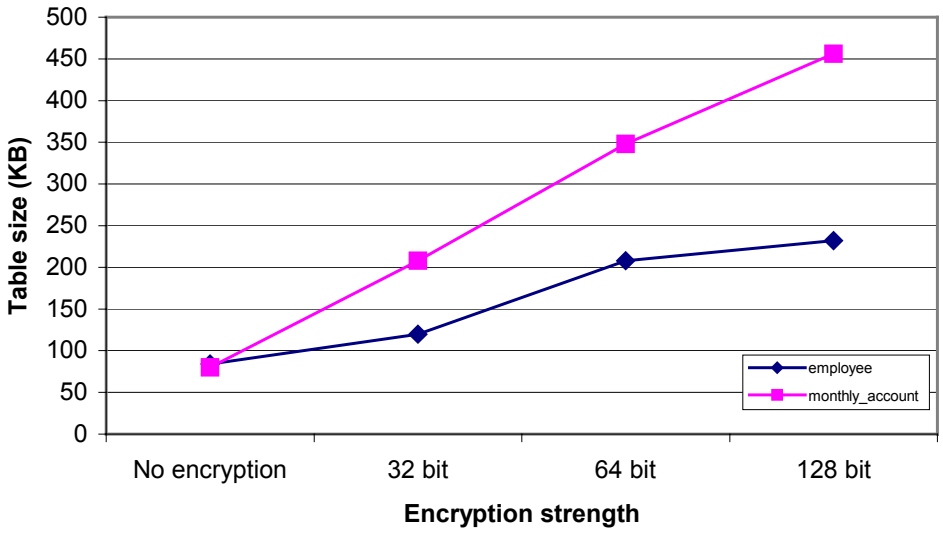


Figure 3-9: Table size with regard to encryption key length

Both Figure 3-8 and Figure 3-9 suggest that the time and space that the customer needs to create the encrypted tables at the service provider's site depend heavily on the number of encrypted attributes and on the key length. However, Figure 3-7 suggests that once the tables are created, the service performance time does not suffer. In the proposed system, manipulating data is costly while querying data is fast. These figures only give a trend on computational sensitivities and do not claim to meet all practical requirements such as minimal key length, etc.

3.8.3 Practical implementation issues

The implementation at hand is based on a JAVA applet that performs encryption and decryption as well as the post-processing on the client side. The applet would be loaded by the client every time the service is requested.

A more efficient approach would require the service provider to deliver a certified *browser plug-in,* which contains the transformation scheme and needs to be installed and parameterized by the client. The latter includes creation of a secret key. Sensitive data to be transmitted would then be marked with a specific HTML tag that forces the plug-in to

encrypt the information before sending it.

Enterprise solutions could eventually take advantage of a proxy server through which every IP packet needs to pass. The proxy could check every packet for marked-up sensitive data and, if applicable, would transform the tag's content. See Figure 3-10 for an illustration of this. An advantage of this setup is that the secret key is only kept at the proxy and not on every individual customer's machine.

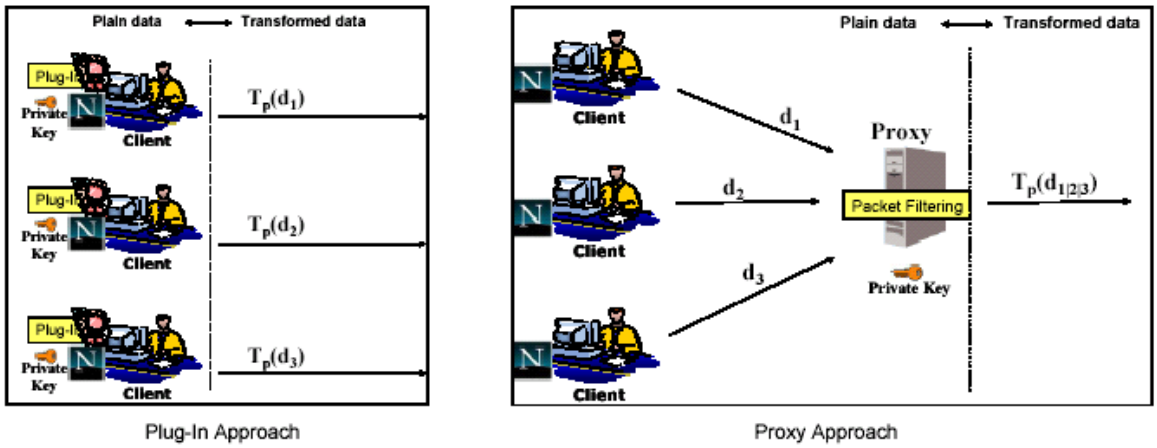


Figure 3-10: Implementation via plug-ins (left) and via proxy server (right)

Both approaches assume the existence of locally installed browsers. In the future, this may not always be necessary, as new techniques like the *remote GUI* only require the presentation layer to be processed at the customer site. With the data management completely shifted to the central facility, transforming sensitive information must then take place at the (untrusted) service provider location.

3.9 Limitations and opportunities

The proposed architecture should not be considered as a "one-size-fits-all" solution that works for every kind of network services. It focuses on applications that require some basic database and arithmetic operations on sensitive data. It is especially valuable for service bundles whose main value lies in their variety and their completeness in many smaller, granular services. In other words, wage accounting services are more suited for the architecture than for complex data mining metrics such as the ones proposed in [Cutler and Sterne, 2000].

Different attributes require different encryption measures. The proposed algorithm is best suited for numbers that will later be processed with arithmetic operators. Primary key values in contrast often just serve as identification means, and are not subject to later processing. Hence it would be useful to encrypt these values with a very secure

cryptographic algorithm such as *RSA* [Rivest, et al., 1978b]. As the private key *p* has already been chosen, it can be used to encode the key values with this alternate algorithm.

Regarding this, the proposed software solution is not suited for complex calculation problems, but for the aggregation of many single, rather simple services. A good example is the calculation of aggregate HR figures that we discussed here. Speaking in terms of the trade-off idea sketched in the introduction, the trade-off in this approach is that for a certain kind of security, the user can only have a limited number of services *S(D)* from the service provider. He has to give up a certain amount of security to obtain a more extensive service offering.

Complementary to the practical reasoning, there is also theoretical work especially on the question whether or not SQL algebra can be securely outsourced. At least a part of the SQL algebra can be mapped to logical operations such as OR, AND, NOT. However, there does not yet exist a homomorphic bit-encryption scheme that is homomorphic to the complete set of these logical operations [Maurer, 2004]. But even if such an encryption scheme existed, it is not sure whether the entire SQL algebra could be mapped to these logical operations. The results that [Fischmann and Günther, 2003] derive from the related code obfuscation approach by [Barak, et al., 2001] suggest that mapping SQL algebra on encrypted data is not entirely possible. Further research is needed to (a) determine whether the complete set of logical operations can be covered by a homomorphic bit-encryption scheme and to (b) determine which parts of the SQL algebra can be mapped to logical operations.

4 Protecting sensitive information in data for public use

The state has no business in the bedrooms of the nation.
(Pierre Elliott Trudeau, Canadian Prime Minister)

4.1 Motivation and running example

4.1.1 2-party case vs. 3-party case

Chapter 3 dealt with the 2-party case where a holder of confidential data submits confidential data to a service provider who returns the service result. The 3-party case is different in the sense that the data holder is not necessarily the service user at the same time. Instead, the service result is forwarded to a 3rd party that is not necessarily trusted (see the dashed line in Figure 4-1 and Section 2.1).

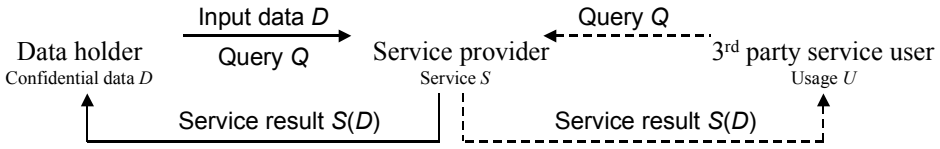


Figure 4-1: 2-party case vs. 3-party case (dashed line)

The existence of an untrusted third party implies that the service result *S(D)* has to be created in such a way that no information about the confidential data *D* can be inferred. If the service provider itself is not trusted, the confidential data *D* has also to be protected from the service provider.

4.1.2 Running example: Regional health initiatives

Regional healthcare initiatives have recently been created to improve the quality of healthcare in their communities by analyzing community-wide healthcare trends and publicizing problematic results. Among other reasons, this development is driven by high numbers of hospital-acquired (*nosocomial*) infections and by increasing hospitalization rates for people with chronic diseases such as diabetes. Figure 4-2 shows this situation for the state of Pennsylvania, USA, a development that has alarmed the healthcare community. For the five-year period, the hospitalizations for diabetes as the principal

diagnosis accounted for over 614,000 hospital days and incurred almost $1.6 billion in hospital charges [PHC4, 2002].

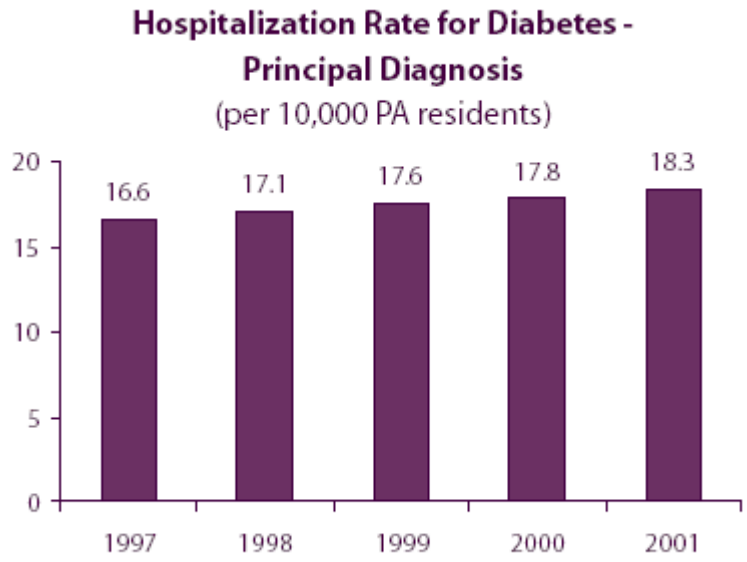


Figure 4-2: A driver underlying the creation of healthcare initiatives
Source: [PHC4, 2002]

Another aim of a regional healthcare initiative is to increase the use of preventive screenings among all affected patients by compiling and releasing information about compliance rates. With regard to diabetes, it is widely believed that adequate diagnostic and preventive measures help reduce short-term complications. Therefore, an important indicator for adequate care is the participation of affected patients in the following preventive screenings.

- A blood test for Hemoglobin A1c (HbA1c) to check a patient's sugar control.
- A measure of a patient's LDL cholesterol levels to prevent heart disease.
- Urinalysis to indicate possible kidney problems.
- Eye exams to prevent glaucoma and retinopathy.
- Foot exams.

Each screening is required to be undertaken on a recommended schedule (e.g. eye exams once a year). Measuring compliance rates is a difficult task because generally, the delivery of healthcare services involves many different parties such as physicians, pharmacies, laboratories, and insurers such as health maintenance organizations (*HMOs*) [Rindfleisch, 1997]. Hence information about patients, disease diagnosis, medications, prevention, and treatment methods is often distributed among heterogeneous databases.

The integration of these heterogeneous data sources with the objective of supporting community-wide data access is an important problem and has been addressed by a number of researchers (see e.g. [Berndt, et al., 2001; Kossmann, 2000; Krishnan, et al., 2001; Wiederhold, 1993]). Approaches range from the creation of data warehouses (see work on CATCH, a data warehouse supporting public health by [Berndt, et al., 2001]) to the use of mediator-based architectures [Wiederhold, et al., 1996]. In this thesis, we focus on mediator-based approaches although all presented methods work for data warehouses too (see Section 4.1.3). Whereas cost and security considerations have usually been taken into account in prior work on mediators, we are more concerned with the privacy implications that can be an outcome of the (desired) data linkage and data fusion enabled by the mediator. Referring to our diabetes case, the involved parties may have different concerns with possible outcomes of the analysis.

- The patient may principally be afraid of a central pooling of her data because the disclosure of a formerly unknown disease might adversely affect life insurance premiums.
- The physician may be confronted with the fact that his test compliance rates differ significantly among patients of different age, race, income, gender, and insurance plan.
- The HMO may fear that detailed internal data may be inferred by competitors and used in marketing campaigns.
- A laboratory may be uncomfortable with the fact that its test analysis times differ significantly among HMOs (although the same fee is charged).

See Figure 4-3 for an illustration of the issue.

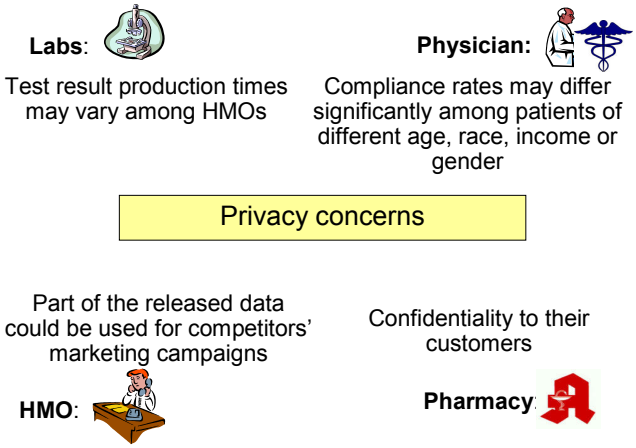


Figure 4-3: Privacy concerns of data providers for chronic disease reports

This chapter analyzes the problem of ensuring that the data released by the mediator (i.e.,

such as one run by the regional healthcare initiative) in support of community health does not permit inferential disclosure of information that is private and confidential. In particular, we focus on the interval inference problem for sensitive HMO data. To illustrate the relevance of this problem, consider the following information about test compliance rates in 2001. It is in part based on real-world data taken from [PHC4, 2002].

TEST	AVERAGE COMPLIANCE AMONG HMOS	STANDARD DEVIATION
HBA1C CHECK	83%	5.7%
LIPID PROFILE	54%	4.7%
EYE EXAM	45%	2.0%

Table 4-1: Compliance rates for diabetes test

HMO	AVERAGE PERFORMANCE
HMO_1	58%
HMO_2	65%
HMO_3	60%
HMO_4	60%

Table 4-2: Average test compliance rates for different HMOs

The upper table contains the mean test compliance rates in the entire community (e.g., a county) and its associated standard deviation. The lower table indicates the general performance of each HMO. Since each HMO considers its own compliance rates for each of these tests, e.g. the HbA1c check, as sensitive data, this information is not displayed. However, given the aggregate data published by the mediator in both tables, bounds can be inferred about the sensitive values. For example, HMO_1 can use its knowledge of its own compliance rates and the published data to infer that HMO_2's compliance rate for the HbA1c check is between 87.2% and 88.5% which corresponds to an inferred interval of [0.872; 0.885]. In Section 4.3.2, we will show which techniques a snooper can deploy to

infer such tight intervals.

Mediators should detect and limit this type of privacy breach. Our objective is to develop new models and methods for the prevention of interval inference that can be incorporated into the mediator.

4.1.3 Data warehouse and mediator architectures (Information integration)

As the information for health reports comes from many different sources that can be very heterogeneous in structure, the data-disseminating institution has to undertake efforts for information integration. Problems that have to be resolved include the heterogeneity of data schemas, differences in time and unit measures as well as country/state-specific characteristics. In particular, we elaborate on the differences between data warehouses and mediator architectures that can both be used for the creation of chronic disease reports.

A *data warehouse* is “a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision making process" [Inmon, 1996]. It has copies of information from several sources stored in a single database and global schema with materialized and/or computed data (a.k.a. *materialized integration*). These data may be preprocessed and are only updated indirectly through the sources. The main advantages of a data warehouse include short response times, powerful query opportunities and high quality of the provided information.

In contrast, a *mediator* stores no data. It is software that supports a virtual database (as if it were materialized) and translates queries that are sent to the mediator into source queries (a.k.a. *virtual integration*). It then synthesizes results and returns an answer to a user query [Wiederhold, 1993]. The main advantages of a mediator include the currency of the data, its small size and a lack of complexity of the system.

There are several approaches for a security-aware mediator such as the ones proposed in [Rezgui, et al., 2002; Wiederhold, et al., 1996], but these approaches are not aware of the privacy implications that can be an outcome of the (desired) data linkage and data fusion enabled by the mediator. Figure 4-4 shows the basic idea of such a mediator.

A medical researcher poses a query to a system of source databases DB_1, DB_2, DB_3,… from physicians, labs, HMOs etc. These sources can be very heterogeneous with regard to structure (relational database management system vs. XML file system) and semantics (e.g. for schema conflicts). Besides the resolution of these technical problems, the mediator also has to protect the confidential data that is contained in the source databases. The core component of such a security mediator is the *query-rewriting* process

[Rezgui, et al., 2002]. An incoming query is screened for sensitive attributes to which the query-issuer has no access rights. All of these confidential attributes are then removed from the query which may even lead to a rejection of the query (e.g., when the query issuer has no access right to the only attribute that he requested).

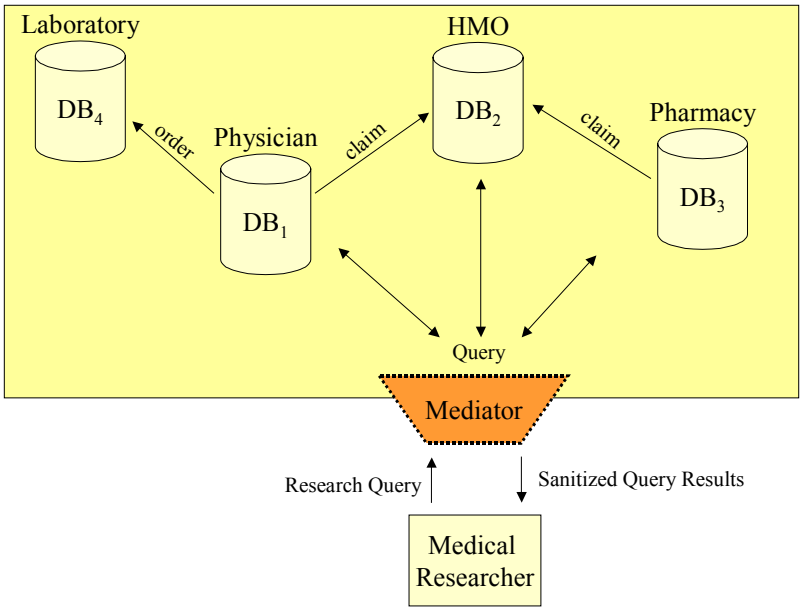


Figure 4-4: A security mediator for healthcare

We will show in Section 4.2 that even if an effective query-rewriting process is in place, confidential information can still be inferred by cleverly combining the uncritical information that the query-issuer actually has access to.

4.1.4 Trust in the mediator

The methods and models that we are going to present in the following sections are based on the assumption that the mediator can be trusted, e.g. as a trusted third party. However, this may not necessarily be the case for the same reasons that a the service provider was not trusted in the two-party case, see Section 3.2. Such a scenario requires the employment of enhanced cryptographic techniques such as secure multiparty computation [Goldreich, 1998]. But even then, it is still questionable whether or not a mediator can deliver useful results. Because one of the major threats is the snooping of one data holder at the other (e.g. when one HMO snoops at the other), we cannot assume trust among the data holders. This makes the use of e.g. homomorphic encryption difficult because the data holders are unlikely to share a common private key.

4.2 Inference problems

4.2.1 Inference control in statistical databases

There are institutions whose only purpose is to collect and analyze data from different sources, to store them in so-called *statistical databases* and to publish excerpts of these databases to the public. Such institutions include Census Bureaus and the mentioned health initiatives. Statistical databases are read-only in nature and related to both the data warehouse and the mediator approach, see [Shoshani, 1997] for an extensive discussion of commonalities and differences. The problems we discuss in the following sections have first been introduced in the context of statistical databases in the 1980s, but can readily be transferred to data warehouse and mediator systems (that have been introduced in the 1990s).

4.2.2 Exact, statistical and interval inference

When a snooper links and analyzes non-critical, public data from different sources to derive estimates for confidential information, we speak of an *inference* of confidential information. If the snooper is able to determine an exact value, we speak of *exact inference*. If the snooper is able to determine a statistical estimator, we speak of *statistical inference*. These two kinds of inference have been dealt with to a large extent in the literature of statistical disclosure control (see surveys in [Adam and Wortman, 1989; Shoshani, 1982]).

[Li, et al., 2002b] introduce a new kind of inference referred to as *interval inference*, giving the following example. Consider a relation with attributes (`model`, `sale`) where `model` is public and `sale` is confidential. Assume further that (legitimate) sum queries for the accumulated sales of models A and C give 200 and for models A and B 4200. Based on this information, we can conclude the following. $A \in [0; 200]$, $B \in [4000; 4200]$ and $C \in [0; 200]$. That means that we can determine the value of B with a maximal error of 200, corresponding to equal or less than 5%.

Strictly speaking, interval inference is a special case of statistical inference where, in this example, the probability $P(4000 \leq B \leq 4200)$ is 100%. To determine the bounds of B in the example above, we had to solve the following two simple optimization problems.

min	B	max	B
s.t.:	A+C= 200	s.t.:	A+C= 200
	A+B= 4200		A+B= 4200
	$A, B, C \geq 0$		$A, B, C \geq 0$

This is a *linear programming problem* (*LP*) with three variables (A, B and C). These

optimization problems can become almost arbitrarily complex when the number of variables is higher and when the type of optimization problem changes (i.e. to *non-linear programming problem* (*NLP*) where at least one constraint is not linear). This would be the case if for example, we were also provided the standard deviation of A, B and C. The constraint would be $(^1/_3((A-\mu_{ABC})^2+(B-\mu_{ABC})^2+(C-\mu_{ABC})^2)^{½} = \sigma_{ABC}$, where σ_{ABC} is given and μ_{ABC} is $^1/_3(A+B+C)$.

4.3 Model and definitions

4.3.1 A two-dimensional table model

We will now introduce the more complex case of a two-dimensional table, where the marginal information such as a metric of central tendency or a measure of dispersion are public, and where the inner cells are confidential.

j (attr.) / i (data holders)				CENTRALITY	DISPERSION
	a_{11}	…	a_{1n}	$a_{1, cen}$	$a_{1, dis}$
	…	…	…	…	…
	a_{m1}	…	a_{mn}	$a_{m, cen}$	$a_{m, dis}$
CENTRALITY	$a_{cen, 1}$	…	$a_{cen, n}$		
DISPERSION	$a_{dis, 1}$	…	$a_{dis, n}$		

Table 4-3: Confidential inner cells and public marginal information

A malicious snooper can use this modeling technique to condense information from two one-dimensional tables into a single table such as Table 4-3. Of course, a snooper can even establish tables with more than two dimensions, see [Chowdhury, et al., 1999; Duncan, et al., 2004]. For reasons of simplicity and graphical intuition, we will stick to the 2-dimensional case in this work. However, all the methods presented from here on are also applicable in *n*-dimensional tables with $n \geq 3$. We will, in order to protect confidential data, use the modeled behavior of a potential snooper in order to check whether a published report leaks confidential information.

Example: We use the published information from our running example (Table 4-1 and Table 4-2) to clarify this approach. The inner cells are now the compliance rates for specific tests of individual HMOs. HMOs might fear that competitors could misuse these data e.g. for marketing campaigns, and thus they consider these data to be confidential. Imagine an HMO who advertises its cancer prevention programs but does not yield exceptional test compliance rates for their customers. Table 4-4 displays the data sheet that an anonymous

snooper without any insider knowledge has. The cell embraced by the dashed line marks a cell that the snooper is interested in; in this case it is the HbA1c test compliance rate of HMO_2.

j (attr.) / i (data holders)	HbA1c	Lipid profile	Eye exam	MEAN μ	STANDARD DEV. σ
HMO_1	a_{11}	a_{12}	a_{13}	58.0%	÷
HMO_2	a_{21}	a_{22}	a_{23}	65.0%	÷
HMO_3	a_{31}	a_{32}	a_{33}	60.0%	÷
HMO_4	a_{41}	a_{42}	a_{43}	60.3%	÷
MEAN μ	83.0%	54.1%	45.4%		
STANDARD DEVIATION σ	5.7%	4.7%	2.0%		

Table 4-4: Marginal information summarized by an anonymous snooper

4.3.2 Mathematical programming

What a potential snooper can do now is to solve a minimization (min) and a maximization (max) problem to determine bounds on the confidential values that he is interested in. Equation (4-1) shows the generalized form of these optimization problems. $G_i(\mathbf{a}) \leq g_{i,\,row}$ denote the constraints imposed by the table rows, $G_j(\mathbf{a}) \leq g_{col}$ denote the constraints imposed by the columns.

$$
\begin{aligned}
&\text{min / max} && a_{i^*j^*} \\
&\text{s.t.:} && G_{i,\,row,\,centrality}(\mathbf{a}) \leq g_{i,\,row,\,centrality},\ i=1..m,\ \mathbf{a} \in \mathbf{R}^{m\times n} \\
& && G_{i,\,row,\,dispersion}(\mathbf{a}) \leq g_{i,\,row,\,dispersion},\ i=1..m,\ \mathbf{a} \in \mathbf{R}^{m\times n} \\
& && G_{j,\,col,\,centrality}(\mathbf{a}) \leq g_{j,\,col,\,centrality},\ j=1..n,\ \mathbf{a} \in \mathbf{R}^{m\times n} \\
& && G_{j,\,col,\,dispersion}(\mathbf{a}) \leq g_{j,\,col,\,dispersion},\ j=1..n,\ \mathbf{a} \in \mathbf{R}^{m\times n} \\
& && a_{ij} \geq 0,\ i=1..m,\ j=1..n
\end{aligned}
\qquad \textbf{(4-1)}
$$

Example: Our running example contains the row mean, the column mean and the column standard deviation. We further assume that from practical experience, a snooper can say that a test compliance rate never drops beneath 30%. To determine the HbA1c test compliance rates for HMO_2, the snooper has to solve the following two optimization problems.

$$
\begin{aligned}
&\text{min/max} && a_{i^*j^*} && \\
&\text{s.t.:} && a_{11}+a_{21}+a_{31}+a_{41} = 4\,a_{cen,1} && \text{Column Means} \\
& && a_{12}+a_{22}+a_{32}+a_{42} = 4\,a_{cen,2} && \\
& && a_{13}+a_{23}+a_{33}+a_{43} = 4\,a_{cen,3} && \\
& && a_{11}+a_{12}+a_{13} = 3\,a_{1,cen} && \text{Row Means} \\
& && a_{21}+a_{22}+a_{23} = 3\,a_{2,cen} && \\
& && a_{31}+a_{32}+a_{33} = 3\,a_{3,cen} && \\
& && a_{41}+a_{42}+a_{43} = 3\,a_{4,cen} && \\
& && \sqrt{\frac{1}{4}\sum_{i=1}^{4}(a_{i1}-\mu_1)^2} = a_{dis,1} && \text{Column standard deviations} \\
& && \sqrt{\frac{1}{4}\sum_{i=1}^{4}(a_{i2}-\mu_2)^2} = a_{dis,2} && \\
& && \sqrt{\frac{1}{4}\sum_{i=1}^{4}(a_{i3}-\mu_3)^2} = a_{dis,3} && \\
& && 0.3 \le a_{ij} \le 1,\ i=1\ldots4,\ j=1\ldots3 &&
\end{aligned}
\qquad \textbf{(4-2)}
$$

Solving this problem for all cells from a_{11} to a_{43} gives the following intervals.

i (data holders) \ j (attr.)	HbA1c	Lipid profile	Eye exam
HMO_1	[0.74; 0.86]	[0.46; 0.58]	[0.42; 0.49]
HMO_2	[0.84; 0.92]	[0.54; 0.62]	[0.42; 0.49]
HMO_3	[0.74; 0.90]	[0.46; 0.61]	[0.42; 0.49]
HMO_4	[0.75; 0.90]	[0.46; 0.61]	[0.42; 0.49]

Table 4-5: Inferred intervals for the initial running example

Note that these derived intervals have lengths as small as 0.07, which may be a significant compromise of the confidential numerical values. We will now show how the data holders can spell out a level of protection and how the mediator uses these levels to protect confidential data.

4.3.3 Privacy protection policies

When the mathematical model is built and the inferred intervals are determined, we have to check whether the privacy concerns of the data holders are not compromised, i.e.

whether or not the inferable intervals are too tight. Therefore, the data holders (in our running example: the HMOs) have to state their privacy protection policies.

Definition (privacy protection policy):

A *privacy protection policy* for a number of confidential numerical values a_{ij} , is equivalent to protection intervals $[l_{ij}; u_{ij}]$ that are built around the confidential values a_{ij}. For reasonably chosen privacy protection policies $a_{ij} \in [l_{ij}; u_{ij}]$ holds.

This means that every provider of confidential numerical data specifies a tolerance interval around his confidential value that no snooper should be able to "break" in the sense that he cannot tell with 100% certainty that the confidential value lies within the protection interval. The protection intervals are given to the mediator who is in charge of the enforcement of the privacy policies. To the public, of course, the protection intervals are kept secret.

Example: HMO_2 wants to protect its compliance rates with HbA1c tests, lipid profiles and eye exams. We assume the confidential values to be 87.3%, 59.9% and 47.8%, respectively. HMO_2 can build a protection interval with a tolerance level of ±10%, i.e. from $0.9a_{ij}^*$ to $1.1a_{ij}^*$ of the confidential values a_{ij}^*. Table 4-6 depicts HMO_2's confidential values and the protection intervals for different tolerance levels.

Compliance \ Test	HbA1c	Lipid profile	Eye exam
Exact compliance rate	0.873	0.599	0.478
Tolerance level 5%	[0.83; 0.92]	[0.57; 0.63]	[0.45; 0.50]
Tolerance level 10%	[0.78; 0.96]	[0.54; 0.66]	[0.43; 0.53]
Tolerance level 15%	[0.74; 1.00]	[0.51; 0.69]	[0.41; 0.55]

Table 4-6: Protection intervals for HMO_2

Note that we use the term "privacy protection policy" in the context of interval inference. Of course privacy protection policies can be deployed in many other contexts to protect private information. Referring to Table 2-2, a user of web-based services can protect himself against tracking of his behavior by rejecting cookies (consult Section 2.4.2). Every user can implement this policy easily via the privacy preferences of his web browser.

4.3.4 Insider threats

So far, the snooper was assumed to be anonymous with no access to confidential information. Now consider the case where HMOs are snooping at one another, e.g. when

HMO_1 snoops on HMO_2. The underlying information basis for HMO_1 is then broader than the one displayed in Table 4-4, because HMO_1 knows its own test compliance rates, see Table 4-7.

i (data holders) \ j (attr.)	HbA1c	Lipid profile	Eye exam	MEAN μ	STANDARD DEVIATION σ
HMO_1	75.0%	56.0%	43.0%	58.0%	÷
HMO_2	a_{21}	a_{22}	a_{23}	65.0%	÷
HMO_3	a_{31}	a_{32}	a_{33}	60.0%	÷
HMO_4	a_{41}	a_{42}	a_{43}	60.3%	÷
MEAN μ	83.0%	54.1%	45.4%		
STANDARD DEVIATION σ	5.7%	4.7%	2.0%		

Table 4-7: Underlying information for snooping HMO_1

Knowing this, the optimization models that the snooping HMO must build and solve are different from the anonymous case as depicted below.

$$
\begin{array}{lll}
\text{min/max} & a_{i^*j^*} & \\
\text{s.t.:} & a_{11}+a_{21}+a_{31}+a_{41} = 4\ a_{cen,1} & \text{Column Means} \\
 & a_{12}+a_{22}+a_{32}+a_{42} = 4\ a_{cen,2} & \\
 & a_{13}+a_{23}+a_{33}+a_{43} = 4\ a_{cen,3} & \\
 & a_{11}+a_{12}+a_{13} = 3\ a_{1,cen} & \text{Row Means} \\
 & a_{21}+a_{22}+a_{23} = 3\ a_{2,cen} & \\
 & a_{31}+a_{32}+a_{33} = 3\ a_{3,cen} & \\
 & a_{41}+a_{42}+a_{43} = 3\ a_{4,cen} & \\
 & \sqrt{\frac{1}{4}\sum_{i=1}^{4}(a_{i1}-\mu_1)^2} = a_{dis,1} & \text{Column standard deviations} \\
 & \sqrt{\frac{1}{4}\sum_{i=1}^{4}(a_{i2}-\mu_2)^2} = a_{dis,2} & \\
 & \sqrt{\frac{1}{4}\sum_{i=1}^{4}(a_{i3}-\mu_3)^2} = a_{dis,3} & \\
 & a_{11} = 0.75 & \text{Insider constraint} \\
 & a_{12} = 0.56 & \\
 & a_{13} = 0.43 & \\
 & 0.3 \le a_{ij} \le 1,\ i=1\ldots4,\ j=1\ldots3 &
\end{array}
\qquad (4\text{-}3)
$$

Because HMO_1 has more information at hand than the anonymous snooper, the

determination of the confidential values can be much more precise. Table 4-8 shows the results after the solution of the mathematical programming problems above.

i (data holders) \ j (attr.)	HbA1c	Lipid profile	Eye exam
HMO_1	0.75	0.58	0.43
HMO_2	[0.87; 0.89]	[0.59; 0.60]	[0.47; 0.48]
HMO_3	[0.83; 0.86]	[0.48; 0.52]	[0.45; 0.47]
HMO_4	[0.83; 0.87]	[0.49; 0.53]	[0.45; 0.47]

Table 4-8: Inferred intervals by HMO_1 with insider information

If you compare the intervals inferred by HMO_1 with the ones that the anonymous snooper could infer (Table 4-5), the reduction of the interval length is significant. Thus, it is very important to consider the insider threat when analyzing data for public use. A "public" user may well be an insider who wants to analyze public data for malicious purposes.

4.3.5 Interval inference

Given the confidential data a_{ij} , the protection intervals $[l_{ij}; u_{ij}]$ and the intervals inferred by the snooper $([a_{ij}^{l}; a_{ij}^{u}])$, we can now formally define the occurrence of interval inference.

Definition (interval inference):

An *interval inference* occurs iff there exists an interval $\boldsymbol{I} = [a_{ij}^{l}; a_{ij}^{u}]$, such that

(i) $\forall x \in [0;1]$: If x proves all column and row constraints, then $x \in \boldsymbol{I}$

(ii) $\boldsymbol{I} \in [l_{ij}; u_{ij}]$

(iii) $a_{ij}^{*} \in \boldsymbol{I}$.

This definition implies that an interval inference only occurs if (and only if) the bounds inferred by the snooper both lie within the protection interval and include the sensitive value a_{ij}^{*}. Note that the methods we are going to present also accommodate other definitions such as the one given in [Li, et al., 2002b]. In Figure 4-5, we see three possible cases. In the upper case, the protection interval is completely covered by the inferred interval and there is clearly no interval inference. This is also the case in the middle, where the lower inferred bound is very close to the confidential value, but still the inferred interval is not completely included by the protection interval. The only critical case is the third one. The deduced interval is entirely included in the protection interval and thus an interval inference occurred. The report that allowed for this inference cannot be published as it is and should be adjusted to better protect the sensitive datum a_{ij}^{*}.

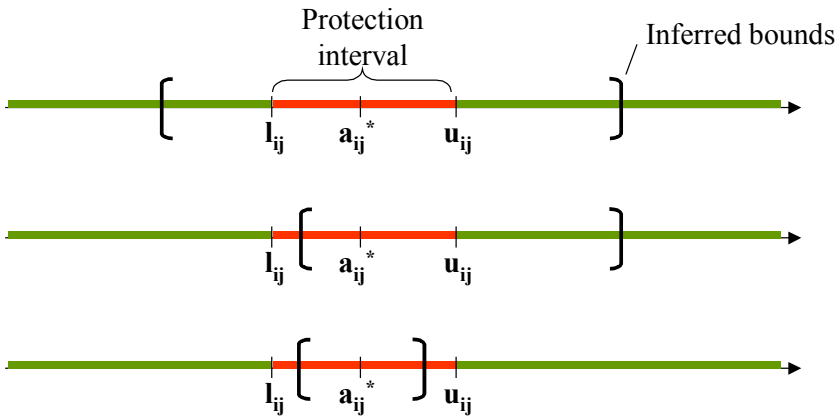


Figure 4-5: Detecting interval inference

Based on this definition, we can now match the intervals inferred by the snooper (Table 4-5) with HMO_2's protection intervals (Table 4-6). The first row shows the intervals that the snooper was able to infer. Rows two to four show the protection intervals at different levels of tolerance. If a cell is marked orange, it indicates an instance where the snooper was able to compromise HMO_2's privacy. The inferred interval is too small to be accepted, and thus the report should not be published as it is.

Compliance \ Test	HbA1c	Lipid profile	Eye exam
Inferred intervals	[0.84; 0.92]	[0.54; 0.62]	[0.42; 0.49]
Tolerance level 5%	[0.83; 0.92]	[0.57; 0.63]	[0.45; 0.50]
Tolerance level 10%	[0.78; 0.96]	[0.54; 0.66]	[0.43; 0.53]
Tolerance level 15%	[0.74; 1.00]	[0.51; 0.69]	[0.41; 0.55]

Table 4-9: Interval inferences at different tolerance levels

Note that the greater the protection interval, the more "cautious" the data holder is and the more interval inferences occur. In the table, a tolerance level of 15% accounts for the compromise of all test rates, but at a tolerance level of 5%, only the HbA1c test rate is compromised.

4.4 Limiting interval inference

We saw in the last section that a data-disseminating institution can assume the role of a snooper in order to check whether the published information complies with the preferences of the data holders. However detecting interval inference is not enough - it also has to be limited. In this section we discuss related work and show why existing approaches have to be extended in order to satisfy our purposes.

The SDC literature basically distinguishes between two different methods to limit interval inference, data perturbation and query restriction [Adam and Wortman, 1989; Agrawal and Srikant, 2000]. Random data perturbation means distorting the original data, storing the modified values in a second database and granting users access only to the perturbated database. Query restriction grants users access to the original database, but strictly monitors and possibly rejects incoming queries. This is shown in Figure 4-6.

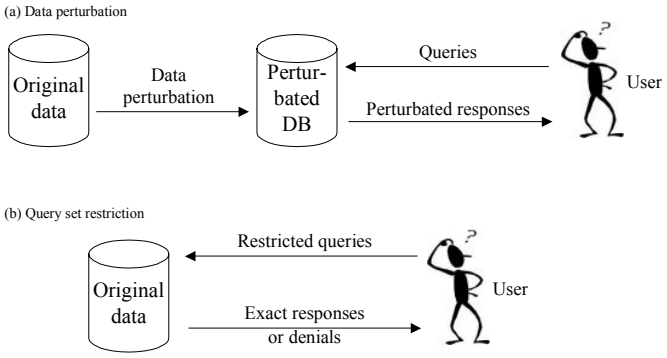


Figure 4-6: Data perturbation (a) vs. query restriction (b)

4.4.1 Data perturbation

As already discussed in Section 2.3.3.2, data perturbation is a technique to perturb the original values such that confidential, individual data become useless for a snooper while the statistical properties of the attribute are preserved. The manipulated data is stored in a second database and is then freely accessible for the users (Figure 4-6 (a)). With regard to interval inference, a new technique called *Random Data Perturbation* has been introduced by [Li, et al., 2002a; Li, et al., 2002b].

This approach deals with inferential disclosure directly by publishing falsified data only. Original values are never published. Instead, a probability distribution is used to change the original values such that they never lie within the bounds of the protection interval.

This probability distribution is called ε-δ-*Gaussian*. It is derived from the standard normal distribution with mean μ, the confidential original value, and standard deviation σ_ε. The protection interval is modeled as $[\mu-\varepsilon; \mu+\varepsilon]$. The probability that the perturbed value (the one that is actually published) lies within this interval is zero (see the red interval in Figure 4-7). Only the green values lie outside the protection interval and yield sufficient protection for the confidential value, in this case μ.

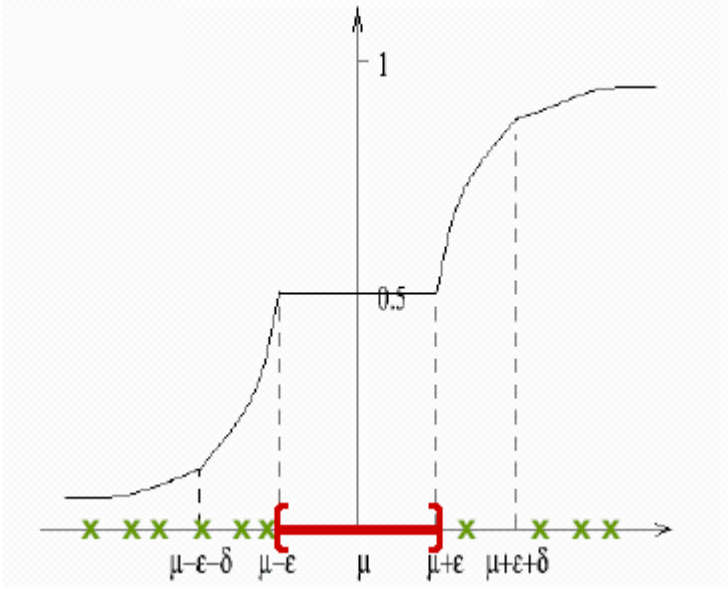


Figure 4-7: Random data perturbation with ε-δ-Gaussian
Source: [Li, et al., 2002a]

If a random draw gives a value x that lies e.g. in $[\mu-\varepsilon; \mu]$, the perturbed value is not $\mu-\varepsilon$ but $\mu-\varepsilon-(\delta/\varepsilon)(\mu-x)$ in order to hide the bounds of the protection intervals. δ determines the length of the stretch to which "red" values are mirrored. The interval $[\mu-\varepsilon; \mu]$ is thus mirrored to $[\mu-\varepsilon-\delta; \mu-\varepsilon]$ which obviously has length δ.

Example: Assume HMO_2's confidential HbA1c test compliance rate is 0.873 and the protection intervals are built according to a tolerance level of 5%, i.e. the protection interval is ([0.83; 0.92]), ε = 0.0437. δ can be chosen arbitrarily to determine the spread of the distribution. We choose δ = 0.1. The Gaussian distribution has mean μ = 0.873. If a random draw from this distribution gives x = 0.85, then we have to shift this value out of the protection interval with the given method. We calculate $x^* = \mu-\varepsilon-(\delta/\varepsilon)(\mu-x)$ = 0.873 - 0.0437-(0.1 / 0.0437)(0.873-0.85) = 0.777. This means that instead of the confidential 87.3%, we publish a value that is guaranteed to lie outside the protection interval, in this case 77.7%

Note that the difference between original and perturbed value is significant. As the aim is to preserve the statistical properties of the attribute, in large data sets this might be balanced. However, in scenarios with small data sets such as the one presented in our running example, the methods turns out to be inefficient.

4.4.2 Query restriction

Opposed to falsifying data, the query restriction technique is concerned with restricting

access to confidential data. [Gopal, et al., 1998] investigate interactive database management systems with an audit approach. Auditing means tracking the queries of a specific user. Before answering a new query from the user, the audit system predicts whether this new piece of information could allow the user to infer confidential information. They present a computationally efficient method to accept or reject queries for sums, maximums and minimums. However, they do not consider nonlinear information such as the standard deviation. Also, the user-based audit does not account for insider threats. If users A and B collaborate, they could pose complementary queries to the system and analyze their common data set.

In this work, we pursue an approach that lies in between query restriction and data perturbation. We do not falsify data but propose a stepwise aggregation of confidential data, e.g. by grouping several values and by publishing aggregate information about this group. Only where absolutely necessary, do we undertake the suppression of information. Therefore, we present an iterative "audit and aggregate" methodology that automatically detects and limits interval inference. It is particularly suited for small data sets such as the one presented in the running example. We are not perturbing data and at the same time, we facilitate the consideration of non-linear marginal information such as the standard deviations displayed in the running example.

4.4.3 Aggregation

Before we introduce the methodology, we draw on an alternative to perturbing data or restricting access to it, the stepwise aggregation of confidential information, see [Li, et al., 2002c; Winkler, 2002]. A well-known form of aggregation is the grouping of values of micro-entities (such as survey respondents) and then publishing information only about the group. With a higher the number of group members the level of aggregation also increases. This aggregation technique can often be found in statistical Census databases, when the entity count in categorical classification is two or less. We give a simple example. In Table 4-10, there is only one Protestant with an income greater than $50,000. This might lead to the re-identification of the respondent in question by using other information not included in the table, see [Willenborg and Waal, 2001]. To solve this, the Census Bureau could therefore take two categories (such as "Protestant" and "Catholic") and unite them into one new group (such as "Christian"). Now, no single category has two members or less.

Religion \ Income	<$5,000	$5000 ≤ x< $10,000	$10,000 ≤ x< $50,000	≥ $50,000
Protestant	27	31	8	1
Catholic	30	34	10	4
Other	48	46	21	7

Religion \ Income	<$5,000	$5000 ≤ x< $10,000	$10,000 ≤ x< $50,000	≥ $50,000
Christian	57	65	18	5
Other	48	46	21	7

Table 4-10: Original (above) and aggregated (below) Census tables

This technique is also employed in many different settings, e.g. when a database table has to be anonymized. To avoid the uniqueness of single respondents (and thereby prevent his or her re-identification), zip codes can be generalized. Figure 4-8 gives two examples of how to aggregate with generalization. The highest level of aggregation is of course "aggregating it all" which means not releasing any details.

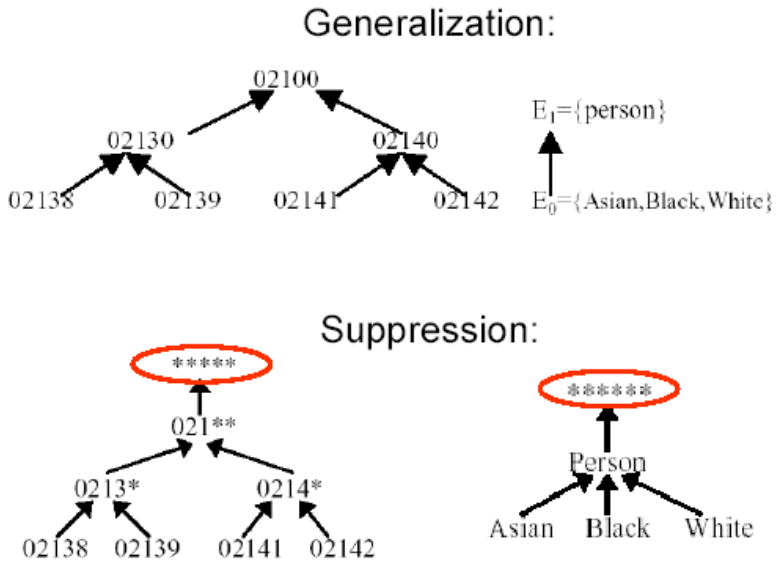


Figure 4-8: Generalization and suppression for purposes of aggregation

For numerical data, aggregation can also take place on a pure structural level. In Table 4-11, we depict marginal information such as measures for central tendency and measures of dispersion. For measuring dispersion, there exist several metrics such as variance, standard deviation, mean absolute deviation or the spread between minimum and maximum values. These metrics contain a different complexity of information. For an attribute of *n* values (e.g. the number of preventive diabetes tests), the standard deviation

$\sigma_i = (\frac{1}{n}\sum_j (a_{ij} - \mu_i)^2)^{\frac{1}{2}}$ is calculated based on *n* values (a_{i1} to a_{in}). In contrast the spread $s_i = \max_j \{a_{ij}\} - \min_j \{a_{ij}\}$ is based only two values. That means that the spread potentially yields less information to the service user. At the same time, the number of variables that a potential snooper has information about also decreases (from *n* to 2). With regard to central tendency, the situation is comparable. The arithmetic mean includes again *n* values, whereas e.g. the median is only a single value.

Utility \ Measure	CENTRAL TENDENCY (MC)	DISPERSION (MD)
High [3]	Arithmetic mean	Standard deviation
⇓ aggregate	$\mu_i = \frac{1}{n}\sum_j a_{ij}$	$\sigma_i = (\frac{1}{n}\sum_j (a_{ij} - \mu_i)^2)^{\frac{1}{2}}$
Medium [2]	Median	Spread
⇓ aggregate	$m_i = a_{i\lceil N/2 \rceil}'$ (a_{ij}' is ordered in j)	$s_i = \max_j \{a_{ij}\} - \min_j \{a_{ij}\}$
Low [1]	Suppress ÷	Suppress ÷

Table 4-11: Data utility of marginal information

Note that this hierarchy is only an example; moreover, the medium level of data utility does not necessarily yield less information than the high level. Especially when the number of values is very small, the spread can sometimes be more of a precise measure than the standard deviation. But still, the exemplary hierarchy gives a solid idea of how aggregation can be performed based on the structure of the numerical information. The largest aggregation is again the suppression of information, indicated by the data utility "low" in the table.

4.5 The "audit & aggregate" methodology

4.5.1 Data holders' privacy concerns vs. service users' data quality needs

In this section, we introduce a methodology that allows a data-disseminating institution such as a regional health initiative to publish data that considers both the privacy concerns of the data holders and the data quality needs of the service users. Speaking in terms of the terminology introduced in Section 2.1, we want to enable the data-disseminating institution to process the confidential data *D* from the data holders to produce a service result *S(D)* that protects *D* towards the service users (i.e. the medical researchers) while yielding the best service result possible.

We will use the protection policies introduced in Section 4.3.3 to model the privacy concerns of the data holders. Information should only be published if the data-disseminating institution can guarantee that the protection intervals of the data holders cannot be broken by a snooper.

On the other hand, the quality needs of the service users are modeled with the structure of marginal information as displayed in Table 4-11. The lower the level of aggregation applied to the marginal information, the higher the utility to the service users. Again, we consider suppression as the largest kind of aggregation possible and do not list it as a category of its own.

In Chapter 5, we will more extensively elaborate on the trade-off between data privacy for the data holders and data utility to the service users.

4.5.2 Data dissemination strategies and categories of interest

We will now formalize the problem that the data-disseminating institution has to solve within our framework. Given the raw data a_{ij}, and the privacy policies $\mathbf{l}_i$ and $\mathbf{u}_i$ of the data holders, we have to find a permissible solution for publication that suits the service users' needs best. In other words, we have to find a good data dissemination strategy.

Definition (dissemination strategy)

The *dissemination strategy (DS)* chosen by the data-disseminating institution denominates the type of marginal data elements for rows ($\boldsymbol{a_{i,cen}}$, $\boldsymbol{a_{i,dis}}$) and columns ($\boldsymbol{a_{cen,j}}$, $\boldsymbol{a_{dis,j}}$) that are published in the final report and their level of utility to the report users.

Example: In Table 4-1 and Table 4-2, the row information is arithmetic mean and the column information is arithmetic mean and standard deviation. Using the aggregation hierarchy from Table 4-11, we publish the information that is marked up in Table 4-12.

Utility \ Cat. C	Row centrality $\mathbf{a_{i,cen}}$	Row dispersion $\mathbf{a_{i,dis}}$	Col. centrality $\mathbf{a_{cen,j}}$	Col. dispersion $\mathbf{a_{dis,j}}$
High [2]	Arithmetic mean	Standard deviation	Arithmetic mean	Standard deviation
Medium [1]	Median	Spread	Median	Spread
Low [0]	Suppress	Suppress	Suppress	Suppress

Table 4-12: A sample dissemination strategy

This dissemination strategy can also be written in the following way.

(Rows) $a_{*,cen}$ **::** Arithmetic mean [2], $a_{*,dis}$ **::** - Suppression - [0]

(Columns) $a_{cen,*}$ **::** Arithmetic mean [2], $a_{dis,*}$ **::** Standard deviation [2]

In this case, we have a set **C=** $\{a_{*,cen}, a_{*,dis}, a_{cen,*}, a_{dis,*}\}$ of four different information categories. All possible assignments of data utilities (here: [0], [1] or [2]) to information categories define the *search space of dissemination strategies*. The aim of the data-disseminating institution is to find in this space the data dissemination strategy DS^* that both satisfies the privacy concerns of the data holders while at the same time fulfilling the need of the service users. We are going to model the user requirements by introducing *categories of interest* **CI** $\subseteq$ **C** that users can specify and that the data-disseminating institution can set.

4.5.3 An iterative methodology

We propose an iterative methodology that starts from an initial solution and subsequently checks whether the proposed solution complies with the privacy policies of the data holders (*audit*). If this is not the case, we adapt the proposed solution with two methods that we will introduce in the next section (*aggregate*). This procedure is repeated until the privacy protection intervals of the data holders are respected. i.e. a permissible solution is found. Figure 4-9 depicts a sketch of the "audit & aggregate" methodology.

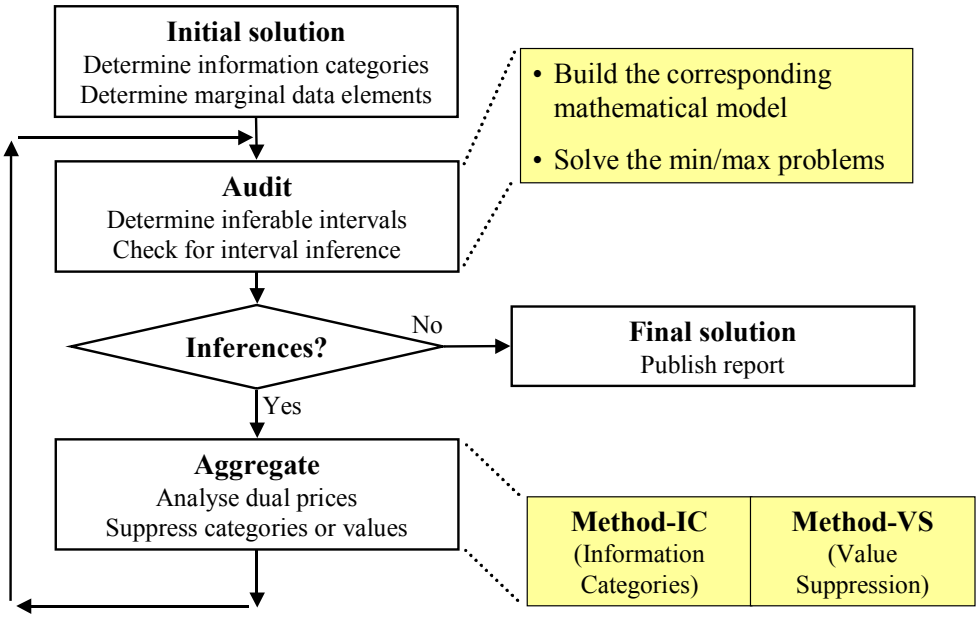


Figure 4-9: The audit & aggregate methodology

Note that both Method-IC and Method-VS are heuristics, i.e. they look for a good solution in the search space of possible published reports. It would also be possible to determine a

global optimum, although this may be time-consuming especially for the case that the aggregation hierarchy for the marginal information is very fine-grained.

The audit is automatically carried out after a new solution is proposed by the aggregation technique. This part of the methodology is discussed using each of the methods in the following sections.

4.6 A method based on choosing information categories (Method-IC)

This method assumes that the search for a data dissemination strategy is purely based on the information categories such as the ones introduced in Table 4-11. That means that if a particular strategy has been found (such as the one in Table 4-12), *all* values for these categories will be published. No single values can be suppressed. If a dissemination strategy fails to pass the audit, an aggregation can only be done by suppressing all values for this category entirely or by changing the type of metric used in this category. This can be done by aggregating information as indicated by the arrows in Table 4-11, e.g. by replacing the column standard deviation by the column min-max-spread.

We assume that for any table, a subset of categories of interest can be specified. For our example this could be **CI=** $\{a_{*,cen}, a_{*,dis}, a_{cen,*}, a_{dis,*}\}$, which would mean measures for row centrality (HMO performance), column centrality (average test compliance) and column dispersion (differences among HMOs). According to the classification in Table 4-11, the initial proposal for a dissemination strategy would be suppression of row dispersion and highest data utility for the remaining (interesting) categories.

4.6.1 Auditing

After specifying this initial solution, we have to run the audit on this first proposal. Therefore, we have to specify a minimization and a maximization problem for every confidential cell and solve them in order to determine the inference intervals. Each dissemination strategy corresponds to a mathematical model that can be automatically generated. Examples of two sample dissemination strategy depicted in Figure 4-10.

In these mathematical programming problems, each class of constraints corresponds to an information category such as the arithmetic means. Each class of row constraints consists of m single constraints, each class of column constraints consists of n single constraints. Every single constraint corresponds to a published marginal value. If we did not publish a specific marginal value (such as e.g. $a_{1,\ cen}$), the snooper could not use this value and would therefore have to suppress the constraint in his mathematical programming problem. This of course widens the intervals that can be inferred on the confidential cells.

	$a_{i,cen}$	$a_{i,dis}$	$a_{cen,j}$	$a_{dis,j}$
[2]	Arithmetic mean	Standard deviation	Arithmetic mean	Standard deviation
[1]	Median	Spread	Median	Spread
[0]	Suppress	Suppress	Suppress	Suppress

corresponds to

min/max $a_{i^*j^*}$

subject to:

$$\frac{1}{n}\sum_j a_{ij} = \mu_i,\ i=1..m$$

$$\frac{1}{m}\sum_i a_{ij} = \mu_j,\ j=1..n$$

$$\left(\frac{1}{m}\sum_i (a_{ij}-\mu_j)^2\right)^{\frac{1}{2}} = \sigma_j,\ j=1..n$$

$0 \le a_{ij} \le 1,\ i=1..m,\ j=1..n$

	$a_{i,cen}$	$a_{i,dis}$	$a_{cen,j}$	$a_{dis,j}$
[2]	Arithmetic mean	Standard deviation	Arithmetic mean	Standard deviation
[1]	Median	Spread	Median	Spread
[0]	Suppress	Suppress	Suppress	Suppress

corresponds to

min/max $a_{i^*j^*}$

subject to:

$$\frac{1}{n}\sum_j a_{ij} = \mu_i,\ i=1..m$$

$$\frac{1}{m}\sum_i a_{ij} = \mu_j,\ j=1..n$$

$0 \le a_{ij} \le 1,\ i=1..m,\ j=1..n$

Figure 4-10: Dissemination strategies and corresponding mathematical programming problems for confidential cells

Solving the mathematical programming problem that corresponds to the initially proposed dissemination strategy gives the inference intervals. Based on the definition of interval inference given in Section 4.3.5, we can now determine the cells that are due to interval inference, i.e. the cells whose inferred bounds are tighter than the protection bounds specified by the data holder.

4.6.2 Aggregation

4.6.2.1 Dual prices

In order to limit the occurrence of interval inference, we will now have a closer look at the critical cells, in particular at the optimization problems that we had to solve during the audit. In the optimal solution of each problem, we can determine the *dual price DP* (a.k.a. *shadow price*) of each active constraint. The dual price of a constraint denominates the hypothetical increase in the objective function when the right-hand side of the constraint is increased by one unit [Winston, 1991]. If a constraint has a dual price of zero, then a relaxation of this constraint does not yield any change for the value of the objective function. All other constraints that have a dual price greater than zero are called *active* constraints. Each constraint is induced by a published marginal value. Therefore, we can denote $DP_{i,\ cen,\ max}$ ($a_{i^*j^*}$) as the dual price that results from the constraint induced by the

marginal value $a_{i,cen}$ in the maximization problem for confidential cell $a_{i^*j^*}$. The denotations for marginal values $a_{i,dis}$, $a_{cen,j}$ and $a_{dis,j}$ as well as for minimization are corresponding follow the same pattern and are determined the same way.

A constraint with high dual price indicates that a relaxation of this constraint would increase the value of the objective function of the maximization problem to a high extent. The same accounts for the decrease of the corresponding minimization problem. As we determine intervals based on a min- and a max-problem, the relaxation of a constraint with high dual price implies a large widening of the inferred intervals. If our aim was to increase the width of the inferred intervals by the suppression of a single class of constraints, we would choose the one with maximum dual price. In each optimization problem, each class of row constraint (such as row arithmetic mean) has *m* instances (one for each published marginal value in the row), each class of column constraint has *n* instances (one each column). We therefore calculate the average dual prices $ADP(a_{i^*j^*})$ for each class of constraint for each confidential cell value $a_{i^*j^*}$.

$$ADP_{*,cen}(a_{i^*j^*}) = \frac{1}{2m}\sum_{i=1}^{m}(DP_{i,cen,\min}(a_{i^*j^*}) + DP_{i,cen,\max}(a_{i^*j^*})) \qquad \text{(row centrality)}$$

$$ADP_{*,dis}(a_{i^*j^*}) = \frac{1}{2m}\sum_{i=1}^{m}(DP_{i,dis,\min}(a_{i^*j^*}) + DP_{i,dis,\max}(a_{i^*j^*})) \qquad \text{(row dispersion)}$$

$$ADP_{cen,*}(a_{i^*j^*}) = \frac{1}{2n}\sum_{i=1}^{n}(DP_{cen,j,\min}(a_{i^*j^*}) + DP_{cen,j,\max}(a_{i^*j^*})) \qquad \text{(column centrality)}$$

$$ADP_{dis,*}(a_{i^*j^*}) = \frac{1}{2n}\sum_{i=1}^{n}(DP_{dis,j,\min}(a_{i^*j^*}) + DP_{dis,j,\max}(a_{i^*j^*})) \qquad \text{(column dispersion)}$$

As we have already executed the audit, we know which confidential cells are subject to interval inference. This allows us to put aside the cells that are already protected. We can now focus on the critical cells and we can calculate the critical average dual price *CADP* for each class of constraint. We use the indicator variable $x_{i^*j^*} = \begin{cases} 1, & a_{i^*j^*}\ critical \\ 0, & a_{i^*j^*}\ uncritical \end{cases}$ to indicate whether or not a confidential cell $a_{i^*j^*}$ was subject to interval inference. ω denotes the number of critical cells. This gives the following values for *CADP*.

$$CADP_{*,cen} = \frac{1}{\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*} ADP_{*,cen}(a_{i^*j^*}) \qquad \text{(row centrality)}$$

$$= \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}\sum_{i=1}^{m}(DP_{i,cen,\min}(a_{i^*j^*}) + DP_{i,cen,\max}(a_{i^*j^*}))$$

$CADP_{*,dis}$ $= \frac{1}{\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*} ADP_{*,dis}(a_{i^*j^*})$ (row dispersion)

$$= \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}\sum_{i=1}^{m}(DP_{i,dis,\min}(a_{i^*j^*}) + DP_{i,dis,\max}(a_{i^*j^*}))$$

$CADP_{cen,*}$ $= \frac{1}{\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*} ADP_{cen,*}(a_{i^*j^*})$ (column centrality)

$$= \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}\sum_{j=1}^{n}(DP_{cen,j,\min}(a_{i^*j^*}) + DP_{cen,j,\max}(a_{i^*j^*}))$$

$CADP_{dis,*}$ $= \frac{1}{\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*} ADP_{dis,*}(a_{i^*j^*})$ (column dispersion)

$$= \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}\sum_{j=1}^{n}(DP_{dis,j,\min}(a_{i^*j^*}) + DP_{dis,j,\max}(a_{i^*j^*}))$$

During the audit we observe the *CADP* for those information categories that were published and search for the maximum *CADP*.

Example: During the interval inference detection for the initial solution of our running example, we obtained the following critical average dual prices. Note that a measure for row dispersion was not published, and thus did not impose any constraints on the optimization problems

Row centrality: $CADP_{*,cen}$ = 1.02

Row dispersion: $CADP_{*,dis}$ = 0 (not published)

Column centrality: $CADP_{cen,*}$ = 1.77

Column dispersion: $CADP_{dis,*}$ = 29.19 ← Max.

The information category "column dispersion" has the highest critical average dual price. This means that this class of constraints has the greatest tightening impact on the inferred intervals.

We pick the maximum *CADP* because this information category has the highest impact on the width of the inferred intervals. The hypothesis is that if we suppress this information category or at least reduce the amount of information delivered in this category, we can

significantly widen the intervals that are inferable by a snooper.

In any case, as at least one of the confidential cells is subject to interval inference, we have to adapt the data dissemination strategy. We propose using the hierarchy presented in Table 4-11 and to reduce the data utility in the information category with the highest *CADP*. In the case of our running example, this means that we reduce the data utility in column dispersion from [2] to [1], see Table 4-13.

	$a_{i,cen}$	$a_{i,dis}$	$a_{cen,j}$	$a_{dis,j}$
[2]	Arithmetic mean	Standard deviation	Arithmetic mean	Standard deviation
[1]	Median	Spread	Median	Spread
[0]	Suppress	Suppress	Suppress	Suppress

▶

	$a_{i,cen}$	$a_{i,dis}$	$a_{cen,j}$	$a_{dis,j}$
[2]	Arithmetic mean	Standard deviation	Arithmetic mean	Standard deviation
[1]	Median	Spread	Median	Spread
[0]	Suppress	Suppress	Suppress	Suppress

Table 4-13: Adaptation of the dissemination strategy

The new resulting dissemination strategy is the following.

(Rows) $a_{*,cen}$ **::** Arithmetic mean [2], $a_{*,dis}$ **::** Suppression [0]

(Columns) $a_{cen,*}$ **::** Arithmetic mean [2], $a_{dis,*}$ **::** Spread [1]

We can now run the audit again. The adaptation of the dissemination strategy is repeated until a permissible solution is found. This method always terminates, i.e. finds a permissible solution because in the worst case, all values for the categories of interest will be suppressed. The optimization problems would thus only have the non-negativity constraints. We have an *unconstrained optimization problem* of the following form.

$$\text{min/max} \qquad a_{i^*j^*}$$

$$\text{subject to:} \qquad 0 \leq a_{ij} \leq 1,\ i=1..m,\ j=1..n$$

This optimization does not give the snooper any information except the fact that the confidential values are between 0 and 1.

The whole procedure is summed up in Figure 4-11. We will further investigate the properties of Method-IC in the experiments that we discuss in Section 4.8.

Method: METHOD-IC

Input: Confidential data a_{ij}. Protection intervals $[l_{ij}; u_{ij}]$. Set of categories of interest **CI**$\subseteq$ **C=**$\{a_{*,cen}, a_{*,dis}, a_{cen,*}, a_{dis,*}\}$.

Output: A dissemination strategy DS in terms of published marginal information categories

Steps

for all C$\in$**CI let** data_utility(C)= 2 // assign highest utility to all categories of interest

for all C$\in$(**C** \ **CI**) **let** data_utility(C)= 0 // assign lowest utility to the remaining categories

repeat

let disclosure_detection = false

for all inner table cells (i,j)

solve_corresponding_(N)LPs // solving the max and min (N)LP for cell a_{ij}

if ($[\min(a_{ij}); \max(a_{ij})] \subseteq [l_{ij}; u_{ij}]$) **and** ($a_{ij}^{*} \in [\min(a_{ij}); \max(a_{ij})]$) // interval inference conditions are fulfilled

let disclosure_detection = true

if (disclosure_detection = true)

let C* = {C$\in$ **CI** | Constraints imposed by C on the (N)LPs have maximum critical average dual price CADP}

decrease(data_utility(C*), 1) // this updates DS by reducing data utility in the category with maximal CADP

until (disclosure_detection = false) // a permissible solution is found

return DS

Figure 4-11: Pseudo-code for Method-IC

4.7 A method based on value suppression (Method-VS)

Method-IC is purely based on the choice of the appropriate information categories, neglecting the ones that the users are not interested in and reducing the data utility only for those categories that yield a significant widening of the inferable intervals.

We now propose a refinement of Method-IC that promises to lose even less information during the "aggregate" step of the "audit and aggregate" methodology. In contrast to Method-IC, we do not assume that *all* values of an information category have to be published. Instead, during the first audit we look for the single marginal value (such as the measure of dispersion for a particular row/column) that most restricts the inferable intervals for critical cells in terms of its critical dual price *CDP*. Instead of changing or suppressing the entire information category, we only suppress the single marginal value with highest critical dual price. If all values of an information category are suppressed and if interval inference still occurs, the principle of Method-IC is applied and we reduce data utility in this category by 1. The initial solution is identical then to the one used in Method-IC.

Note that our optimization criterion does not use the average dual price *ADP* (the one that was used in Method-IC) but the single dual prices *DP*. The critical dual price $CDP_{i,cen|dis}$ for a particular row *i* (or $CDP_{cen|dis,j}$ for a particular column *j*) is thus defined as follows. Again,

we use the indicator variable $x_{i^*j^*} = \begin{cases} 1, & a_{i^*j^*}\ critical \\ 0, & a_{i^*j^*}\ uncritical \end{cases}$ to indicate whether or not a confidential cell $a_{i^*j^*}$ was subject to interval inference and ω to count the number of critical cells.

$$CDP_{i,cen} = \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}(DP_{i,cen,\min}(a_{i^*j^*}) + DP_{i,cen,\max}(a_{i^*j^*}))$$ (centrality in row *i*)

$$CDP_{i,dis} = \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}(DP_{i,dis,\min}(a_{i^*j^*}) + DP_{i,dis,\max}(a_{i^*j^*}))$$ (dispersion in row *i*)

$$CDP_{cen,j} = \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}(DP_{cen,j,\min}(a_{i^*j^*}) + DP_{cen,j,\max}(a_{i^*j^*}))$$ (centrality in column *j*)

$$CDP_{dis,j} = \frac{1}{2m\omega}\sum_{i^*=1}^{m}\sum_{j^*=1}^{n} x_{i^*j^*}(DP_{dis,j,\min}(a_{i^*j^*}) + DP_{dis,j,\max}(a_{i^*j^*}))$$ (dispersion in column *j*)

During the audit we observe the *CDP* for all published marginal values and search for the maximum *CDP*.

Example: During the interval inference detection for the initial solution of our running example, we obtained the following critical dual prices *CDP*. Note that the marginal values for row dispersion were not published and thus did not impose any constraints on the optimization problems. This is why their critical dual price is zero.

i (data holders) \ j (attr.)				$CDP_{i,cen}$	$CDP_{i,dis}$
	a_{11}	a_{12}	a_{13}	1.5	0
	a_{21}	a_{22}	a_{23}	0	0
	a_{31}	a_{32}	a_{33}	1.5	0
	a_{41}	a_{42}	a_{43}	1.5	0
$CDP_{cen,j}$	2.5	1.5	1.5		
$CDP_{dis,j}$	8.0	9.4	22.8	← max	

Table 4-14: Critical dual prices for published marginal data elements

We would now suppress the marginal value that has the highest dual price, in the example the measure of dispersion for column 3. The intermediate dissemination strategy is the following.

(Rows) $a_{*,cen}$ **::** Arithmetic mean [2], $a_{*,dis}$ **::** - Suppression - [0]

(Columns) $a_{cen,*}$ **::** Arithmetic mean [2], $a_{dis,1}$ **::** Standard deviation [2]
$a_{dis,2}$ **::** Standard deviation [2]
$a_{dis,3}$ **::** Suppression [0]

This is the new dissemination strategy that is, again, subject to an audit. A peculiarity occurs when all single values of a specific information category, e.g. all column standard deviations, are suppressed. Instead of running the audit with the information category missing entirely, we reduce the data utility of the entire category by one and publish all single values with the reduced data utility. In Figure 4-12 we can see that after the suppression of the last column standard deviation, we switch to the information category with next lowest data utility. In this case, following the classification in Table 4-11, we switch from standard deviation to min-max spread (cf. Figure 4-12).

i \ j	HbA1c	Lipid profile	Eye exam	μ	σ
HMO_1	a_{11}	a_{12}	a_{13}	58.0%	÷
HMO_2	a_{21}	a_{22}	a_{23}	65.0%	÷
HMO_3	a_{31}	a_{32}	a_{33}	60.0%	÷
HMO_4	a_{41}	a_{42}	a_{43}	60.3%	÷
μ	83.0%	54.1%	45.4%		
σ	***	***	***		

▶

i \ j	HbA1c	Lipid profile	Eye exam	μ	σ
HMO_1	a_{11}	a_{12}	a_{13}	58.0%	÷
HMO_2	a_{21}	a_{22}	a_{23}	65.0%	÷
HMO_3	a_{31}	a_{32}	a_{33}	60.0%	÷
HMO_4	a_{41}	a_{42}	a_{43}	60.3%	÷
μ	83.0%	54.1%	45.4%		
MAX-MIN	12.3%	8.6%	4.8%		

Figure 4-12: Reducing data utility after suppressing all values of a category

"Audit and aggregate" is repeated until either a permissible solution is found or until all marginal values are suppressed. Figure 4-13 shows a sketch of Method-VS.

Elements of Method-IC can be seen in the second case differentiation. When all marginal values in the information category **C*** are suppressed, the data utility in the entire category is reduced by 1. This is what Method-IC does directly after the first interval inference has occurred (for the category with the highest average dual price *ADP*). Figure 4-14 shows the full specification of Method-VS in pseudo-code.

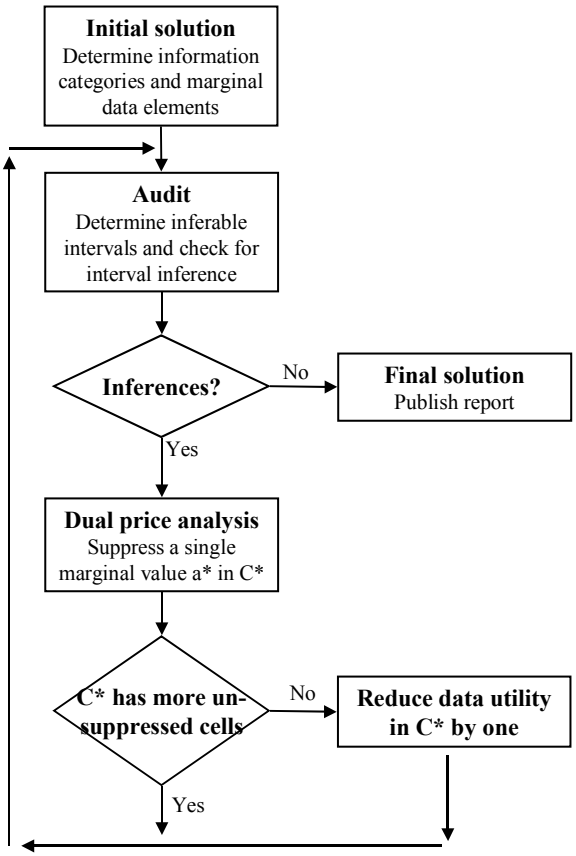


Figure 4-13: A sketch of Method-VS

Method: METHOD-VS

Input: Actual data a_{ij}. Protection intervals $[l_{ij}; u_{ij}]$. Set of categories of interest **CI**$\subseteq$ **C=**$\{a_{*,cen}, a_{*,dis}, a_{cen,*}, a_{dis,*}\}$.

Output: A dissemination strategy DS in terms of published marginal data elements with possibly suppressed single values

Steps

for all C$\in$**CI let** data_utility(C)= 2 // assign highest utility to all categories of interest

for all C$\in$(**C** \ **CI**) **let** data_utility(C)= 0 // assign lowest utility to the remaining categories

repeat

let disclosure_detection = false

for all inner table cells (i,j)

solve_corresponding_(N)LPs // solving the max and min (N)LP for cell a_{ij}

if ([min(a_{ij}); max(a_{ij})] $\subseteq$ [l_{ij}; u_{ij}]) **and** (a_{ij}^{*} $\in$ [min(a_{ij}); max(a_{ij})]) // interval inference conditions fulfilled

let disclosure_detection = true

if (disclosure_detection = true)

let a* = Marginal data element whose related constraint on the (N)LPs has maximum critical dual price CDP

suppress(a*, C*) // this suppresses marginal value a* in C* for its high CDP and updates the DS

if (a* is the last value to be suppressed in its information category C*) //if all values in C* suppressed

decrease(data_utility(C*), 1) // publish all values in C* at a decreased level of data utility

until (disclosure_detection = false) // a permissible solution is found

return DS

Figure 4-14: Pseudo-code for Method-VS

4.8 A prototypical implementation

4.8.1 Goals of the implementation

We implemented both Method-IC and Method-VS as well as the Random Data Perturbation method (*Method-RDP*) by [Li, et al., 2002a] with the following set of objectives.

- Compare the methods with regard to the quality of the disseminated information.
- Analyze the sensitivity of the methods with regard to privacy protection policies
- Give computational trends that are dependent of the table size.
- Give computational trends that are dependent of the skew in the original data.

4.8.2 Sketch of the implementation

Our system is based on the approach of [Wiederhold, et al., 1996] (cf. also Figure 4-4). The core component of our system is the *mediating* party as depicted in Figure 4-15.

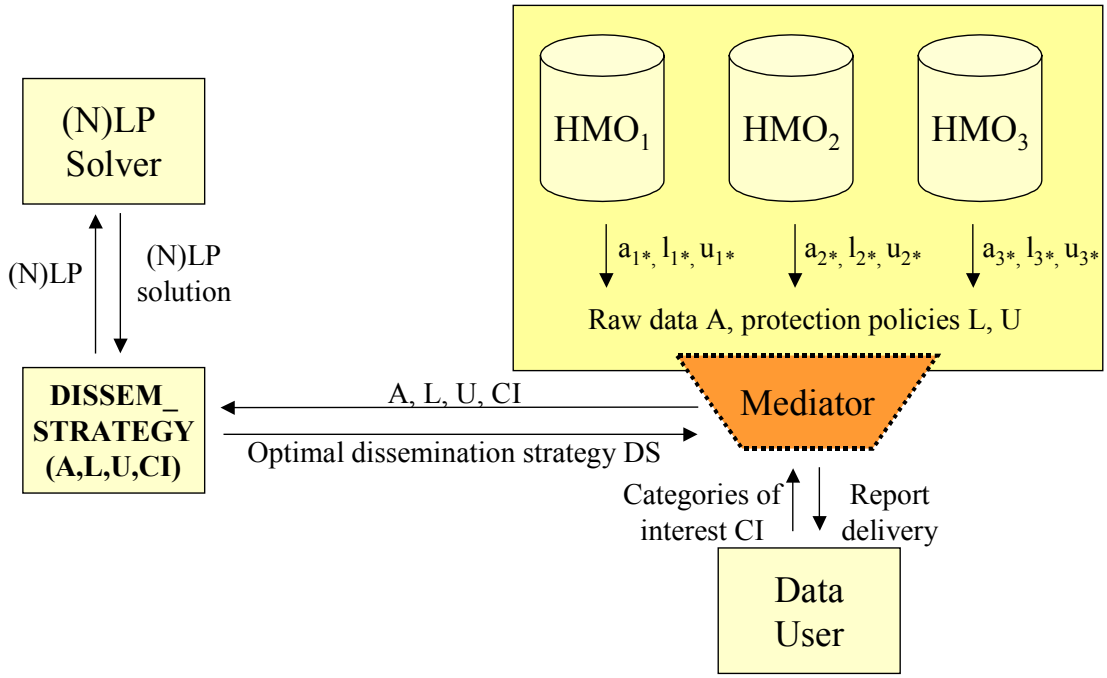


Figure 4-15: Sketch of the implementation

The data holders, in this case the HMOs, provide the raw data a_{ij}, i=1..m, j=1..n as well as their protection policies [l_{ij}; u_{ij}] to the mediator. The service users, in turn, specify their categories of interest *CI*. It is the task of the mediator to take this information and to determine the dissemination strategy that both satisfies the data quality needs of the users and the privacy concerns of the service users.

We are aware of the fact that this implementation is of a purely prototypical character that

has as its sole purpose to prove the suitability of the proposed methods and to explore trends of computational behavior as well as the sensitivity of the model. It needs significant amendments in functional and technological directions in order to cope with the complexity of practical environments.

For our purposes, we used the following technological components for the implementation of this system.

COMPONENT	TECHNOLOGY	REFERENCE
CPU	x86 (700 Mhz)	h18000.www1.hp.com/products/quickspecs/10382_ca/10382_ca.html
RAM	192 MB	
Operating System	MS Windows 2000	http://www.microsoft.com/windows2000
Programming language for Method-IC and for Method-VS	Java 1.4.2	java.sun.com/j2se/1.4.2
Database management system	MS Access 2000	office.microsoft.com/home/default.aspx
Mathematical programming	AMPL	[Fourer, et al., 2003]
LP solver	CPLEX	www.cplex.com
NLP solver	MINOS	http://www.ampl.com/BOOKLETS/ampl-minos.pdf

Table 4-15: Technological components of the implementation

4.8.3 Sensitivity of interval inference with regard to protection intervals

First we take our running example and analyze how the choice of the protection policies influences the number of inferred cells. For reasons of simplicity, we assume the existence of one privacy protection policy for all data holders. This policy in terms of protection intervals is the first parameter that is subject to variation. We chose a set of 4 dissemination strategies.

DS_1: $\boldsymbol{a}_{*,cen}$ **::** Arithmetic mean [2], $\boldsymbol{a}_{*,dis}$ **::** Standard deviation [2], $\boldsymbol{a}_{cen,*}$ **::** Arithmetic mean [2], $\boldsymbol{a}_{dis,*}$ **::** Standard deviation [2]

DS_2: $\boldsymbol{a}_{*,cen}$ **::** Arithmetic mean [2], $\boldsymbol{a}_{*,dis}$ **::** Suppression [0], $\boldsymbol{a}_{cen,*}$ **::** Arithmetic mean [2],

$a_{dis,*}$ :: Standard deviation [2]

DS₃: $a_{*,cen}$ :: Suppression [0], $a_{*,dis}$ :: Suppression [0], $a_{cen,*}$ :: Arithmetic mean [2], $a_{dis,*}$ :: Standard deviation [2]

DS₄: $a_{*,cen}$ :: Arithmetic mean [2], $a_{*,dis}$ :: Standard deviation [2], $a_{cen,*}$ :: Arithmetic mean [2], $a_{dis,*}$ :: Suppression [0]

Figure 4-16 shows these dissemination strategies and the number of inferences according to the protection policy. All dissemination strategies are only permissible if no interval inferences occur at all. Except DS_4, no dissemination strategy is permissible for protection intervals wider than ±5%. DS_1 dominates all other dissemination strategies in the sense that for all possible protection policies, this dissemination strategy always induces the highest number of interval inferences.

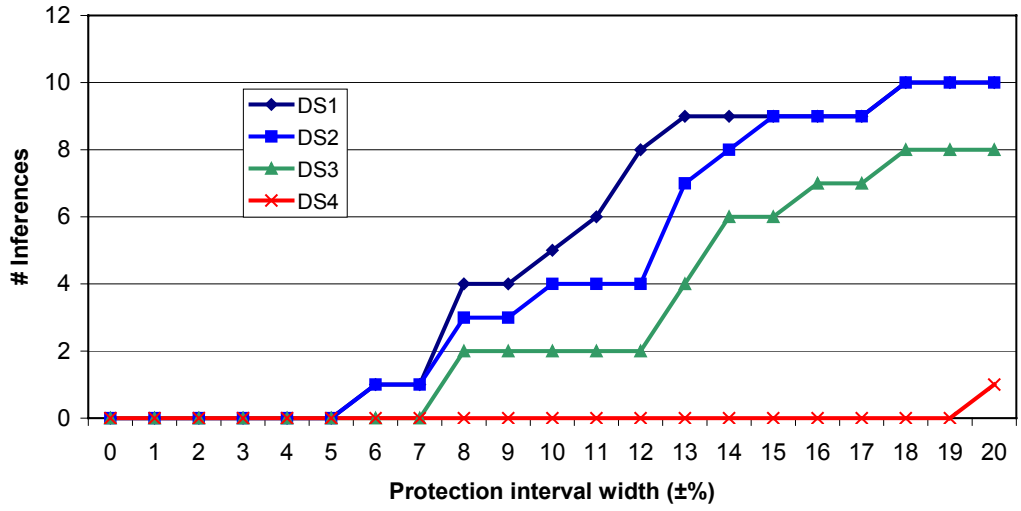


Figure 4-16: Dissemination strategies and inferred intervals for different protection policies

Figure 4-16 indicates that both the privacy preferences of the data holders and the choice of the dissemination strategy have significant impact on the occurrence of interval inferences. We will now examine how these two factors influence the quality of the disseminated information.

4.8.4 Quality of the disseminated information

4.8.4.1 Measuring data quality

Measuring data quality has many different dimensions such as relevance, accuracy, timeliness, accessibility, interpretability and coherence [Ballou and Tayi, 1999;

Brackstone, 1999; Lenz and Rödel, 1991; Naumann, 2002]. Another important issue in database management systems is the problem of missing values, see for example [Harangsri, et al., 1997].

For the purposes of our work, we assume that the underlying raw data complies with all of these quality requirements and that we want to measure the decrease in quality that is induced by the application of our privacy-preserving methods. As the goal of the data-disseminating institution is not publishing microdata but publishing marginal information, we will measure the relative error that a specific disclosure limitation method incurs with regard to the marginal data elements. We define the following error metrics for centrality and dispersion, for columns and rows.

Average relative row error ARE_{row}= $\frac{1}{2m}\sum_{i=1}^{m}\left(\frac{\left|\mu_i-\mu_i^*\right|}{\mu_i^*}+\frac{\left|\sigma_i-\sigma_i^*\right|}{\sigma_i^*}\right)$

Average relative column error ARE_{col}= $\frac{1}{2n}\sum_{j=1}^{n}\left(\frac{\left|\mu_j-\mu_j^*\right|}{\mu_j^*}+\frac{\left|\sigma_j-\sigma_j^*\right|}{\sigma_j^*}\right)$

Average relative error in centrality ARE_{cen}= $\frac{1}{2}\left(\frac{1}{m}\sum_{i=1}^{m}\frac{\left|\mu_i-\mu_i^*\right|}{\mu_i^*}+\frac{1}{n}\sum_{j=1}^{n}\frac{\left|\mu_j-\mu_j^*\right|}{\mu_j^*}\right)$

Average relative error in dispersion ARE_{dis}= $\frac{1}{2}\left(\frac{1}{m}\sum_{i=1}^{m}\frac{\left|\sigma_i-\sigma_i^*\right|}{\sigma_i^*}+\frac{1}{n}\sum_{j=1}^{n}\frac{\left|\sigma_j-\sigma_j^*\right|}{\sigma_j^*}\right)$

These measures only consider "slices" of the table. In order to calculate the overall quality of the disseminated information we can calculate the Total average relative error

$TARE$= $\frac{1}{4}\left(\frac{1}{m}\sum_{i=1}^{m}\left(\frac{\left|\mu_i-\mu_i^*\right|}{\mu_i^*}+\frac{\left|\sigma_i-\sigma_i^*\right|}{\sigma_i^*}\right)+\frac{1}{n}\sum_{j=1}^{n}\left(\frac{\left|\mu_j-\mu_j^*\right|}{\mu_j^*}+\frac{\left|\sigma_j-\sigma_j^*\right|}{\sigma_j^*}\right)\right)$.

If, in the information category of centrality, the arithmetic mean is reduced to the median, we estimate $\mu_i \approx m_i = a_{i\lceil N/2\rceil}$ for the error determination. If in the information category of dispersion, standard deviation is reduced to min-max spread, we can estimate

$\sigma_i \approx \max_j a_{ij} - \min_j a_{ij}$ (4-4)

with

σ_i = $\sqrt{\frac{1}{n}\sum_{j=1}^{n}(a_{ij}-\mu_i)^2} \le \sqrt{\frac{1}{n}\sum_{j=1}^{n}(\max_j a_{ij}-\min_j a_{ij})^2}$

= $\sqrt{(\max_j a_{ij}-\min_j a_{ij})^2} = \max_j a_{ij}-\min_j a_{ij}$

and an incurred relative error of $\frac{\sqrt{\frac{1}{n}\sum_{j=1}^{n}(\max_j a_{ij}-\min_j a_{ij})^2}-\sqrt{\frac{1}{n}\sum_{j=1}^{n}(a_{ij}-\mu_i)^2}}{\sqrt{\frac{1}{n}\sum_{j=1}^{n}(a_{ij}-\mu_i)^2}}$

= $\sqrt{\frac{\sum_{j=1}^{n}(\max_j a_{ij}-\min_j a_{ij})^2}{\sum_{j=1}^{n}(a_{ij}-\mu_i)^2}}-1$

for the information category of, in this case, row dispersion.

If any single marginal value is missing because of suppression, we assume a null value that induces a relative of error 100%. In the case of row standard deviations, this corresponds to $\frac{\left|\sigma_i-\sigma_i^*\right|}{\sigma_i^*}=\frac{\sigma_i^*}{\sigma_i^*}=1$.

We will now run "audit & aggregate" on different sets of information.

4.8.4.2 Method-IC and Method-VS vs. RDP

For the running example, we start with the same initial solution $DS_{initial}$: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Suppression [0], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Standard deviation [2]

for both Method-IC and Method-VS. Again, we analyze privacy protection policies from the range of ±0% (no concerns) up to ±20% (very cautious). Method-RDP disturbs the original data from very small protection intervals on. This rules out any occurrence of interval inference. In turn, high rates of relative errors are incurred. For "low-concern" policies of up to ±5%, we do not observe interval inference either for Method-IC or for Method-VS (the outcome of the audit is obviously the same as the initial solution is identical for both methods). The first interval inference occurs at a protection interval level of ±6%. The methods now react differently.

Method-IC determines the information category that has highest average dual price *ADP*, which in this case is $ADP_{dis,*}$. The data utility for the measure of column dispersion is reduced by 1 from standard deviation to max-min-skew, according to Table 4-11. The estimates for σ_j , according to equation (4-4), give an average relative error for dispersion of ARE_{dis} = 1.20 which corresponds to a total relative error of *TARE*= 0.36. However, the audit on this solution still yields one interval inference, so we have to reduce data utility further. Again, the measure of column dispersion has the highest average dual price $ADP_{dis,*}$. Following Table 4-11, we now have to suppress all marginal values of column dispersion. This solution does not yield any interval inferences anymore. The corresponding error rates are ARE_{dis} = 1 which corresponds to a total relative error of *TARE*= 0.3.

Method-VS instead looks for the single marginal value that has the highest critical dual price, in this case $CDP_{dis,3}$.= 22.79. This is sufficient to limit the occurrence of interval inference. It corresponds to an average relative error of dispersion of ARE_{dis} = 0.33 and a total relative error of *TARE*= 0.1 (cf. Figure 4-17).

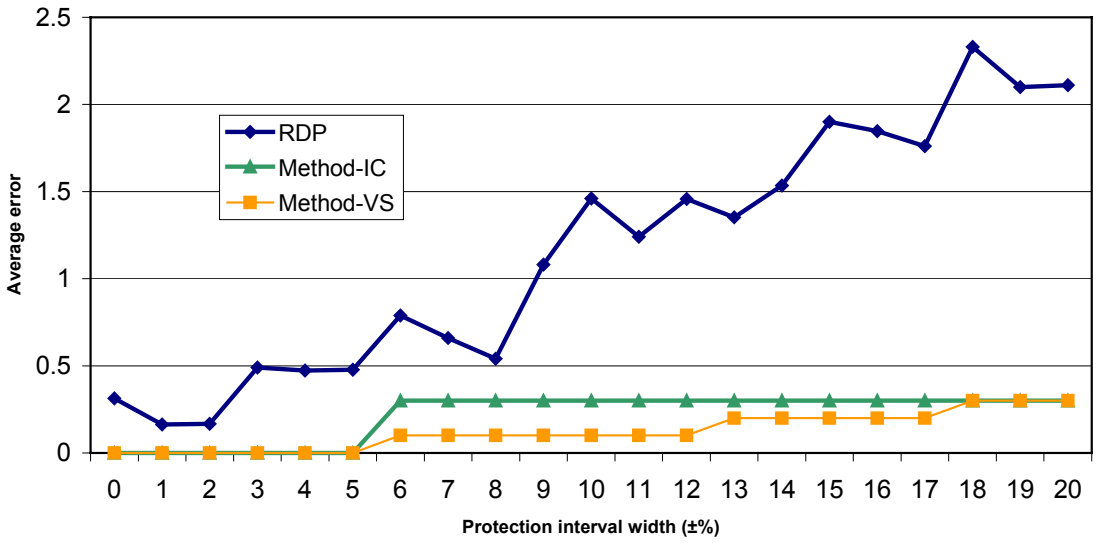


Figure 4-17: Total average relative error (TARE)

We can see already that at low protection intervals, Method-RDP induces errors that both Method-IC and Method-VS can avoid. Also, the relative error of RDP increases on a much higher scale than do Method-IC and Method-VS. For RDP, this increase does not necessarily have to be monotone, i.e. $\exists\ k \in [0.01..0.2]$ with *TARE*(*k*) < *TARE*(*k-0.01*). The reason for this is that the random draws can differ significantly and that a "bad draw" of many values on one particular side of the interval can occur at high protection intervals, too. The relative errors of RDP decrease significantly when the number *mn* of cells increases. Our experiments show that for smaller-sized tables, Method-IC and Method-VS

deliver better results.

Method-IC differs from Method-VS in the sense that it directly changes entire information categories when interval inferences occur. Method-VS, however, only suppresses one marginal value at a time, thus the increase in the total average relative error *TARE* is smaller than with Method-IC. Figure 4-17 illustrates this at for a protection interval of ±6%. Method-IC reduces the data utility in column dispersion in two steps from [2] to [0], whereas Method-VS achieves privacy protection by suppressing only one specific value of column dispersion. This explains the lower total average relative error *TARE*.

The average relative column error ARE_{col} confirms the results obtained for *TARE*. As most of the disclosure limitation is performed within the marginal column values, the main difference is in the scale of the error, as shown in see Figure 4-18.

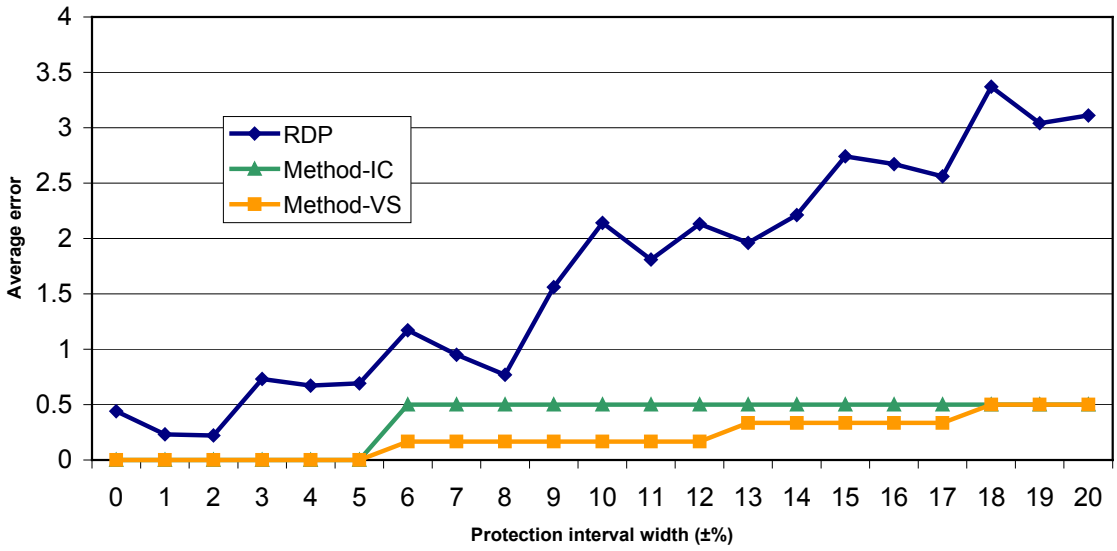


Figure 4-18: Average relative column error (ARE_{col})

We obtained similar results for tables of different size and for raw data that was randomly created with a predefined skew.

4.8.5 Sensitivities of interval inference with respect to table size and skew

Another objective of the prototypical implementation was to determine sensitivities regarding the size of the underlying table and regarding the skew of the raw data.

4.8.5.1 Table size vs. number of inferred cells.

In this section, we measure how the size of the tables influences the number of inferred cells when all other parameters are fixed. For this purpose, we run audits on randomly created tables with given rates of skew and privacy protection policies. The audit

determines the number of inferred cells ω. To make this number comparable between tables of different size, we take the ratio of inferred cells $\omega/(mn)$. A value of 1 means that all cells were subject to interval inference, a value of 0 indicates that no interval inference occurred at all. We randomly created tables of size 2x2, 3x3, 4x4, 5x5 and 6x6, and we looked at protection policies of ±5%, ±10%, ±15%, and ±20%. The other parameters were fixed as follows.

Dissemination strategy: DS_1: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Standard deviation [2], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Standard deviation [2].

The raw data was drawn from a uniformly distributed random variable X with $P(0.6 \leq X \leq 0.8)=1$. The width of this interval is the parameter we will allow to vary in the next section.

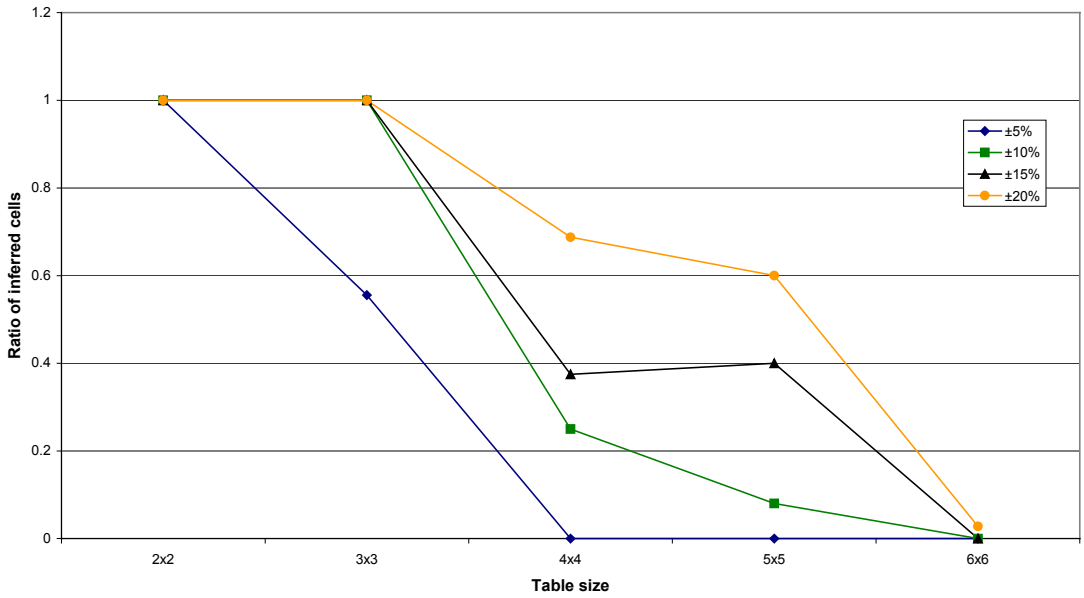


Figure 4-19: Table size vs. ratio of inferred cells for different protection policies

As Figure 4-19 shows, the ratio of inferred cells does not only depend on the width of the protection intervals (as already shown in Figure 4-16), but also largely on the table size. The larger the table, the relatively less interval inferences occur. This underlines the importance of detecting limiting interval inference in settings with a limited number of data holders.

4.8.5.2 Skew vs. number of inferred cells

In this section, we analyze how the skew in the confidential raw data influences the number of inferred cells. As fixed parameters, we chose a 4x4 table with protection intervals of ±15%. We ran tests on the following three dissemination strategies.

DS_1: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Standard deviation [2], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Standard deviation [2]

DS_2: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Min-max skew [1], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Min-max skew [1]

DS_3: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Min-max skew [1], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Suppression [0]

For these dissemination strategies, we obtained the following inference ratios for skews of 5%, 10%, 15%, 20% and 25% (cf. Figure 4-20).

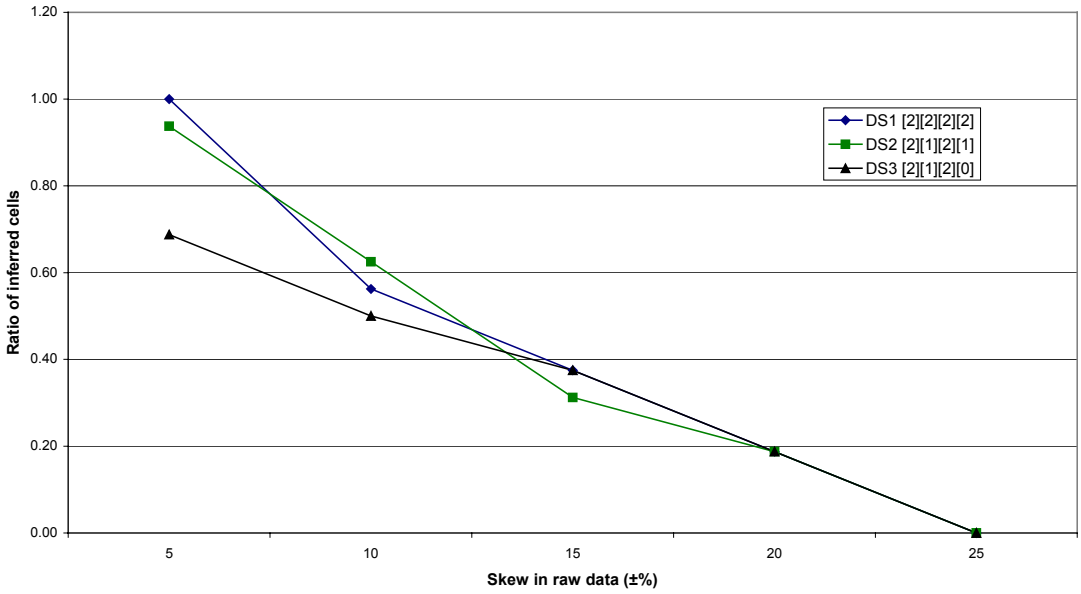


Figure 4-20: Skew in raw data vs. ratio of inferred cells

The lesson taken from this experiment is that the nature of the raw data has a significant impact on the opportunities of a snooper to determine tight intervals for confidential values. If the natural skew of the raw data is great, the relative number of inferred intervals is low. Surprisingly, this also includes dissemination strategies that indicate only little marginal information about dispersion, such as dissemination strategy DS_3 in Figure 4-20.

4.8.6 Complexity

The computational complexity of Method-IC and Method-VS is an important issue, because each audit requires the solution of $2mn$ optimization problems (two for each inner cell) with a maximum of $2(m+n)$ constraints each (m constraints for row centrality, m constraints for row dispersion, n constraints for column centrality and n constraints for column dispersion. We found solutions in reasonable time for table sizes up to 100 inner cells, which is a realistic assumption in the given healthcare setting.

For very large datasets, Method-RDP is potentially better suited, because the creation of perturbed data is easy and because the quality of marginal information gets better with a greater number of datasets. In order to employ "audit and aggregate" in a useful manner, additional efforts have to be undertaken. For example, the complexity of the linear programming problems during the audit can be further reduced by employing graphical cut techniques such as the one proposed by [Gopal, et al., 1998].

4.9 Limitations and opportunities

We proposed an "audit and aggregate" methodology to detect and limit the occurrence of interval inference in distributed database settings. In contrast to the traditional approach of random data perturbation (RDP), we do not falsify raw data but instead drop marginal information for centrality and dispersion where appropriate. For the choice of which data elements to drop, we presented two new methods (Method-IC and Method-VS) that aim at widening the protection intervals to the greatest extent possible while at the same suppressing a minimum amount of information that would be of interest to the service user.

"Audit and aggregate" is particularly suited for small- and medium-sized problems for two reasons. First, the bias in marginal information incurred by Method-RDP is particularly high in these settings. We showed in our experiments that both Method-IC and Method-VS can reduce the incurred relative errors significantly, see e.g. Figure 4-17. Second, the computational complexity of "audit and aggregate" increases significantly for large-scale tables, yet is well controllable for smaller problems. We do not think that this is a severe impediment for practical applications, because the problem of interval inference becomes more and more significant the smaller the problem and the lesser the skew in the original raw data, as described in Section 4.8.5, Figure 4-19 and Figure 4-20.

We are aware of the fact that the presented methods and experiments depend on the choice of marginal information. Additional categories could include a type of total or partial order among row values, e.g. to indicate a performance order among HMOs. Also, user interest is not often easily captured in a single set of categories of interest.

Method-IC and Method-VS are both heuristics that basically search for the best choice within the search space of dissemination strategies. An improvement of "audit and aggregate" could include a complex *meta-optimization* problem that guarantees to find the optimal dissemination strategy in terms of maximal data utility to the users and compliance with the privacy protection policies of the data holders.

5 Privacy trade-offs: Quantitative aspects and implications

Data privacy is necessary. But it should not be misunderstood in a way that it disturbs the activities of the authorities.

(Otto Schily, German Secretary of the Interior)

In the preceding chapters, we have analyzed the trade-off between data holders and service users in two different settings. In the two-party case, the trade-off consisted of obtaining a high level of privacy protection from the service provider at the expense of sacrificing some part of the service offering. The more privacy the data holder wants, the less extensive will the service offering of the service provider be.

In the three-party case, the trade-off consisted of reducing the extent and the quality of the service result in terms of marginal information in order to protect the privacy of the data holders. It was possible to increase the privacy protection for the data holders by reducing the extent of the final service result.

This chapter will elaborate on two important aspects of this trade-off. First, we will discuss quantitative aspects. Which models and metrics exist to quantify the conflict? Which mechanisms can lead to an automatic resolution of the conflict? How can these models and mechanisms be applied to our approaches in the two-party and in the three-party case?

Second, we will discuss the qualitative aspect of the privacy trade-off. Why is it important to raise awareness of the conflict? What can be done to increase data holders' willingness to provide data in well-protected environments? What are the implications for electronic commerce and public policy?

5.1 Quantification

5.1.1 Frameworks in Statistical Disclosure Control

The research field of Statistical Disclosure Control (SDC) has long been concerned with the quantification of the trade-off between data quality for the service users and privacy protection for the data holders. In SDC terminology, the utility to the service users is

measured in terms of *information loss (IL)* compared to the original, unmasked data. The privacy of the data holders is specified in terms of the *disclosure risk (DR)* that indicates the probability with which confidential information can be inferred.

5.1.1.1 Measures for information loss

The aim of SDC research is to find an objective measure for information loss. There exists a seminal debate about the question of whether or not information loss should be specified in objective terms. [Domingo-Ferrer, et al., 2001] argue that the information loss depends heavily on the potential uses of the masked data and that these data uses are "so diverse that it is hard to even identify them". Among other reasons, this is in the implementation of the "audit and aggregate" methodology, we chose a categorical hierarchy of data utilities such as the one displayed in Table 4-11.

[Duncan, et al., 2001a; Duncan, et al., 2001b] propose an information loss criterion for a numerical attribute that assumes value ω with probability p_ω . They use *mean squared precision*, i.e. the reciprocal of the mean squared error. For the case where the actual value ω is equal to 1, the information loss criterion is

$$DU = \frac{\left|dom(\omega)\right|}{\sum_{\omega} p_\omega (\omega - 1)^2} \qquad (5\text{-}1)$$

where |*dom*(ω)| denotes the number of possible values ω.

5.1.1.2 Measures for disclosure risk

Disclosure risk measures the extent to which confidentiality is protected from the attacks of a data snooper. [Duncan, et al., 2001a; Duncan, et al., 2001b] use information theory [Shannon, 1948] to quantify disclosure risk. They suggest the reciprocal of the (non-conditional) entropy, where p_ω = is the probability for the intruder that a cell *X* assumes value ω . The disclosure risk *DR* can then be denoted as

$$DR = \frac{1}{-\sum_{\omega} p_\omega \log p_\omega} \qquad (5\text{-}2)$$

[Domingo-Ferrer and Mateo-Sanz, 2002; Domingo-Ferrer, et al., 2001; Domingo-Ferrer, et al., 2002] note the difficulties in computing p_ω because it is necessary to determine the exact information that a snooper holds. They propose to take the conditional entropy

$$DR(X) = \frac{1}{H(X \mid Y = y)} = \frac{1}{-\sum_{x} p(x \mid y) \log_2 p(x \mid y)} \qquad (5\text{-}3)$$

where *X* is an original cell and *Y* represents the intruder's knowledge (equal to some *y*).

5.1.2 The Risk-Utility confidentiality map

[Duncan, et al., 2001b] were the first to propose a framework that takes both data utility and disclosure risk into account. They plot a *Risk-Utility (R-U) confidentiality map* that includes data points for each disclosure limitation method. With each method, a certain level of disclosure risk is incurred for the data holders and a certain level of data utility is achieved for the service users. There are two extreme cases. First, when all information is suppressed, both data utility and disclosure risk are zero. Second, when all information is published without any transformation, then both data utility and disclosure risk are at their respective maximum. Figure 5-1 shows a sample R-U confidentiality map for a specific disclosure limitation method called *topcoding*.

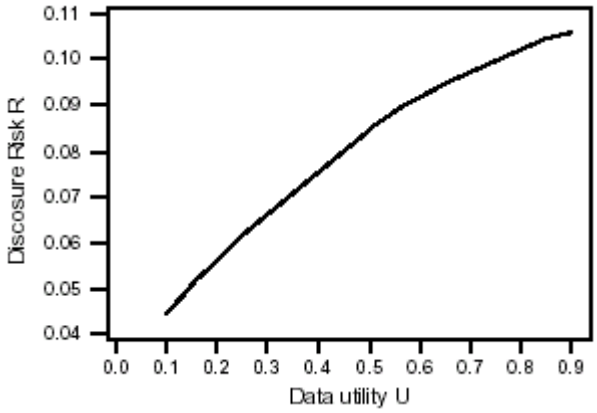


Figure 5-1: The R-U confidentiality map for the disclosure limitation method topcoding with varying parameters
Source: [Duncan, et al., 2001b]

Topcoding is a disclosure limitation method that protects extreme values within in a range of data points. Consider the case of renters in a specific district with their corresponding contract rents. If, for instance, 1% of the renters pay a contract rent of more than $3000, then a topcoding threshold υ = $3000 would suppress the release of individual rents above $3000 and only the number of renters and the mean conditional rent in this segment would be given out. A high υ thus corresponds to a small segment that induces only a little masking. This means high data utility for the users and high disclosure risk for the renters, and is denoted by data points in the upper right of the figure. In contrast, a low threshold υ indicates that the topcoded segment is very large and that many renters are

aggregated in a single value. This means that data utility is low, and that disclosure risk is also minimal. This corresponds to data points in the lower left in Figure 5-1. Note that this map is originally not continuous but a smoothed curve of single data points, i.e. a limited number of topcoding thresholds υ.

5.1.3 A R-U confidentiality map for Health Maintenance Organizations

We will now apply the concept of the R-U confidentiality map to the healthcare example that we have introduced in Chapter 4. We measure the disclosure risk (*DR*) for each confidential cell a_{ij} based on information entropy, as proposed in [Domingo-Ferrer, et al., 2002] where

$$DR(a_{ij}) = \frac{1}{\log_2(1000\left|a_{ij}{}^{u} - a_{ij}{}^{l}\right|)}$$

where $|a_{ij}{}^{u} - a_{ij}{}^{l}|$ is the width of the inferred interval and where 1000 $|a_{ij}{}^{u} - a_{ij}{}^{l}|$ indicates the number of possible values that can be assumed with a precision of 0.001 (i.e. an interval width of 0.015 is equivalent to 15 assumable values). Referring to our running example from Section 4.1.2, we can calculate the total disclosure risk DR_i that a specific health maintenance organization incurs for all of the tests as follows.

$$DR_i = \frac{1}{n}\sum_{j=1}^{n} DR(a_{ij})$$

$$\frac{1}{n}\sum_{j=1}^{n} \frac{1}{\log_2(1000\left|a_{ij}{}^{u} - a_{ij}{}^{l}\right|)}.$$

Measuring data utility is a more difficult issue as it depends on the goals of the service users. For our purposes, we will derive the data utility of a data dissemination strategy by simply adding up the data utilities in the four information categories. The three dissemination strategies that we are going to evaluate for the R-U confidentiality have the following data utilities.

DS_1: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Suppression [0], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Suppression[0]
DU_1= 2+0+2+0= 4

DS_2: $\boldsymbol{a_{*,cen}}$ :: Arithmetic mean [2], $\boldsymbol{a_{*,dis}}$ :: Min-max skew [1], $\boldsymbol{a_{cen,*}}$ :: Arithmetic mean [2], $\boldsymbol{a_{dis,*}}$ :: Suppression[0]
DU_2= 2+1+2+0= 5

DS_3: $a_{*,cen}$ :: Arithmetic mean [2], $a_{*,dis}$ :: Min-max skew [1], $a_{cen,*}$ :: Arithmetic mean [2], $a_{dis,*}$ :: Min-max skew [1]

DU_3= 2+1+2+1= 6

After determining the inference intervals for these three dissemination strategies, we can calculate the disclosure risks and finally draw the R-U confidentiality for the 4 HMOs. This is illustrated in Figure 5-2.

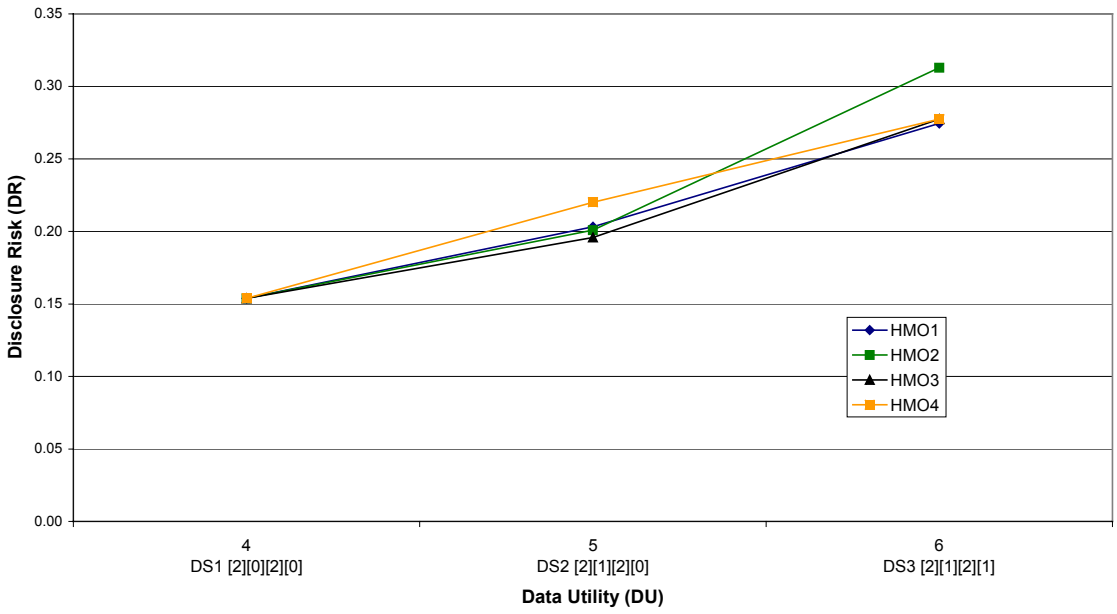


Figure 5-2: R-U confidentiality maps for all HMOs

This figure illustrates the trade-off between data utility and disclosure risk at HMOs. The greater the utility of the disseminated information for the legitimate service user, the greater also the disclosure risk for the data holders. It also indicates that HMOs do not necessarily "dominate" each other in terms of disclosure risk. For DS_2, HMO_2 has higher disclosure risk than HMO_2, but for DS_3 just the opposite is the case. Slight differences in dissemination strategies can have a huge impact on disclosure risk.

Other metrics for data utility are also possible. For example, we could use the reciprocal of the total average relative error *TARE* that we introduced in Section 4.8.4.1. The specification for data utility would shown below.

$$DU' = \frac{1}{TARE} = \frac{4}{\frac{1}{m}\sum_{i=1}^{m}\left(\frac{\left|\mu_i - \mu_i^*\right|}{\mu_i^*} + \frac{\left|\sigma_i - \sigma_i^*\right|}{\sigma_i^*}\right) + \frac{1}{n}\sum_{j=1}^{n}\left(\frac{\left|\mu_j - \mu_j^*\right|}{\mu_j^*} + \frac{\left|\sigma_j - \sigma_j^*\right|}{\sigma_j^*}\right)}$$

The results are comparable to those depicted in Figure 5-2.

5.1.4 Interpretation of the R-U confidentiality map

The R-U confidentiality map does not only illustrate the trade-off between data holders and service users, it is also a good tool to demonstrate how external effects can change the choice of the optimal dissemination strategy. For the R-U map from the last section, we now introduce the concept of a *disclosure risk threshold* that indicates the maximum level of risk that the data holders are willing to incur. If the disclosure risk of a dissemination strategy is above the threshold, it is not permissible, see Figure 5-3.

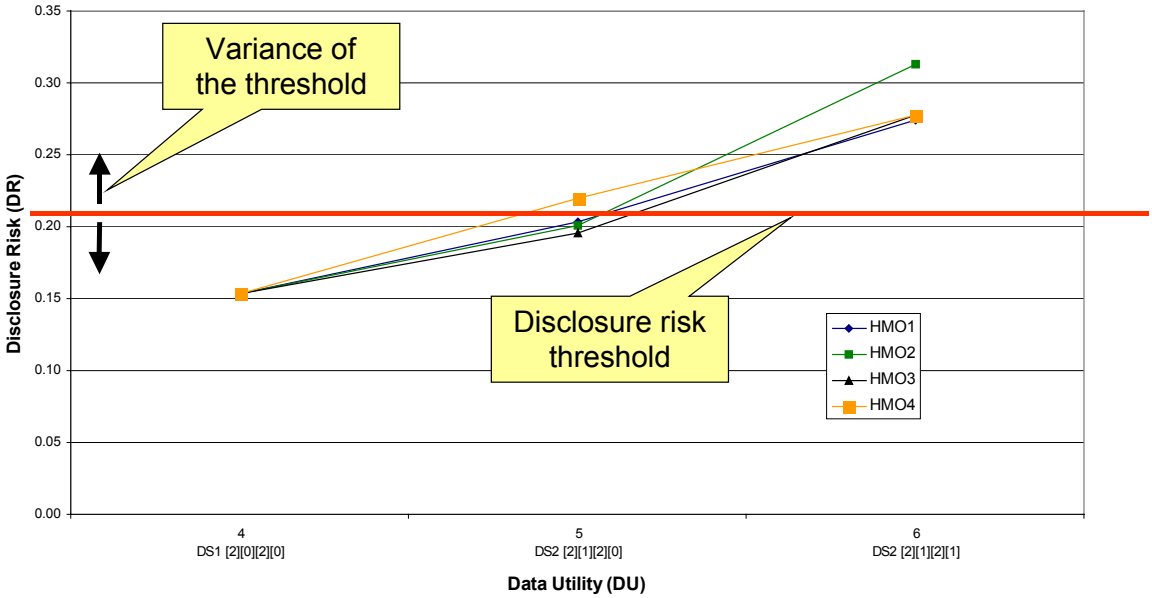


Figure 5-3: Variance of the disclosure risk threshold in the R-U confidentiality map

Now it can easily be seen that for the dissemination strategy DS_2, all HMOs except HMO_4 would agree to a publication. Note that this threshold is not necessarily fixed over time. The most likely reason for a variance in the threshold is a change in the environment of the system. For the case of the HMOs, political or public pressure to improve the cooperation with regional healthcare initiatives can increase the disclosure risk threshold. In the case of national security, which we will address in Section 5.2.2.1, extensive data collection to prevent foreseeable threats may also increase the acceptance of higher exposure of personal data, i.e. a higher disclosure risk threshold.

5.2 Implications

The lack of awareness of the existing privacy trade-off leads to the choice of extremes from the perspective of the data holders. A survey by [Ackerman, et al., 1999] shows that 17% of all online users are "privacy fundamentalists" who will not provide data to a web site even if privacy protection measures are in place. 27% are "marginally concerned" and generally willing to provide data to web sites without major concerns. The remaining 56%

of online users make up the "pragmatic majority" that takes trade-off issues into account and decides on a case by case basis (see Figure 5-4).

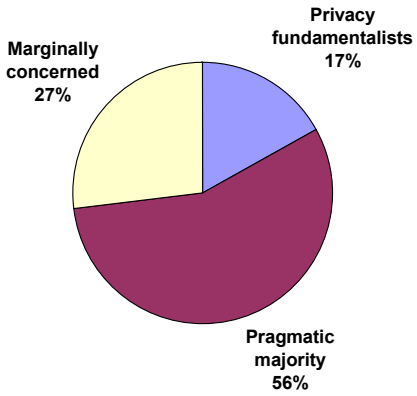


Figure 5-4: Privacy attitudes of online users
Source: [Ackerman, et al., 1999]

A study by [Fox and Rainie, 2000] indicates that the share of privacy fundamentalists could be as high as 27%. When looking at these figures, one has to take into account that the actual usage behavior often differs from the privacy attitudes that were specified through a survey. [Spiekermann, et al., 2001] found that even privacy fundamentalist divulge very private information once they get involved in a particular online process such as an online purchase. However, Figure 5-4 still suggests that almost the half of all online users tend to extreme usage behavior in regards to privacy issues. One reason for this is the lacking awareness of privacy issues and the insufficient belief in privacy protection mechanisms.

5.2.1 Impact on electronic commerce

In spite of growing efforts to protect privacy in web-based services through technical or legal means, many service users are still very concerned about their personal data. A comprehensive study by [IBM, 1999] shows that up to 54% of Internet users in the U.S. have already refrained from buying online due to privacy concerns. In the ASP market, potential customers still rank privacy and security first among their reasons to reject ASP service offerings [Carter, 2000]. Both for corporate and private customers, privacy concerns seem to be a main obstacle for the participation in the electronic marketplace.

One lever to increase market participation is to make the user interaction more transparent. The hypothesis is that if a user knows what is happening with his data, he is more likely to provide them. The *Platform for Privacy Preferences (P3P)* initiative [P3P, 2004] is an "industry standard providing a simple, automated way for users to gain more control over the use of personal information on Web sites they visit" and is accepted by

web sites on an increasing scale [Ernst & Young, 2003]. It automatically matches the privacy preferences of the service user with the privacy policies of the service provider and gives an alert when the user's privacy requirements are not met. It distinguishes between data-oriented policies and method-oriented policies [Kobsa, 2001].

Another approach is to leave the decision about which data to provide directly to the user (instead of an automated privacy negotiation system). This is of particular interest for user-adaptive systems such as online book stores. These systems "cater to users more effectively the more information they possess about them" [Kobsa, 2002]. This means that the more extensive and accurate the information that the system has about the user, the better the service quality that it can deliver. These "personalized services" can include customized finance pages or news collections, targeted recommendations or advertisements based on past purchase behavior, customized pricing, express transactions or tailored email alerts. In online book stores for instance, these personalized services include highlights of recently published books of interest. The recommendations are based on clicking behavior and on preceding book or CD purchases. Unfortunately, many online stores do not offer an option to (de-)activate the tracking of the click and purchase history and thereby prevent the user from easily trading off his own privacy concerns with the potential benefit from an extended service. An interesting approach that considers both privacy concerns and personalization quality can be found in [Berendt and Teltzrow, 2003].

5.2.2 Implications for public policy

The state also plays a major role in the protection of the personal privacy of its citizens. On the one hand, the state is concerned with the enforcement of privacy protection laws such as ones triggered the EU privacy directives [EU, 1995; EU, 2002] for countries of the European Union or the Healthcare Insurance Portability Accountability Act [HIPAA, 1996] in the USA. On the other hand, the state also has an interest in collecting information about its citizens for administrative reasons or for concerns of national security.

5.2.2.1 National security

Intelligence agencies that are charged with crime prevention need to collect information about suspiciously behaving subjects. This may include screening confidential phone calls, emails and conversations of innocent bystanders. On that account, the individual citizen sacrifices a part of his own personal privacy to support higher goals of the society, national security in this case. It is a very difficult political question to decide which goals of the society really justify significant intrusions into the privacy of each individual citizen, see e.g. [Economist, 1999; Orwell, 1949; Schily, 2004; Time, 1997; Warren and Brandeis,

1890]. Reducing the personal privacy of citizens must thus be thoroughly justified. In particular, the effectiveness of the actions taken has to be demonstrated, and the risks for innocent bystanders shall be minimized. It is worth noting that changing political and economic situations have a huge impact on the tolerated level of intrusion. The incidents of September 11, 2001 for instance, have lead to the creation of institutions and laws that allow for a significant invasion of the personal privacy of individuals, see [DHS, 2004; IAO, 2003; PATRIOT, 2001]. The temporarily increased desire for national security leads to a general increase in the tolerance of many citizens to be monitored or to provide personal data. The trade-off between each citizen's privacy and the interests of the service users is settled at a lower level of privacy, triggered by the current political situation. Privacy-defending institutions such as the Electronic Privacy Information Center (EPIC) of course protest against this development, see e.g. [EPIC, 2004a]. In any case, legislation has to ensure that the shifts in political or economic environments neither eliminate the citizens' privacy nor completely block the activity of the authorities.

5.2.2.2 Medical research

There are other domains of public interest. Medical researchers for instance need large samples of patient data to investigate the roots of illnesses such as cancer. Precise data about patient characteristics and behavior facilitate the discovery of correlations between patient characteristics, diagnosis and success of treatment. However, whenever a physician forwards confidential patient data to external third parties, misuse at the expense of the patient must be prevented. Again, the protection of personal privacy is diametrically opposed to the pursuit of a societal goal, in this case the investigation of illnesses to improve society health. Legal frameworks have to ensure that a trade-off can be settled such that both the privacy of the patients is preserved and that the published data has a utility for researchers. In the United States, [HIPAA, 1996] requires the removal of a number of personal data fields before release which, reduces its potential utility for the researchers. There is a lot of work required to facilitate high-quality medical research while still protecting the privacy of the patients [Sweeney, 2002b].

6 Conclusion and future research

Civilization is the progress toward a society of privacy.
(Ayn Rand, American writer)

In this thesis, we analyzed the privacy trade-off between data holders and service users that occurs in various web-based service constructs. An increase in privacy protection for the data holders often implies a decrease in data utility to the service users in terms of extent and precision of the provided data. Our main contributions are as follows.

- A classification of privacy conflicts based on the number of involved parties and the nature of the data provision
- A software-based model for a two-party service architecture where the data holder's privacy is protected at the expense of a restriction in the offered services
- A mediator-based model for a three-party service architecture with a particular focus on the inference of tight bounds for confidential numerical data
- An analysis of different frameworks to quantify the privacy trade-off and an overview on the implications for electronic commerce and public policy.

We classified privacy issues based on two dimensions. The number of involved parties (two vs. three) indicates how many parties have potential insight into the confidential data. In the two-party case, the data holder is the service user at the same time and does not necessarily trust the service provider. A prominent example is the use of financial portfolio services on the web. In the three-party case, the data holder is not necessarily the service user at the same time. A well-known example for this are Census Records where the service result is made available to the public. A second dimension is the nature of the data provision. Personal financial data have to be explicitly specified, whereas personal interests and hobbies can be tracked automatically by web sites who analyze clicking behavior with the help of tools such as cookies.

For the two-party case, we introduced the example of an ASP who offers wage accounting services to its customers. We presented a service architecture that allows the service user to use a limited number of services without submitting plain data to the (potentially untrusted) service provider. A sample implementation suggested that the service performance does not suffer significantly due to the new service infrastructure. The trade-

off in this case consists of the fact that the service user receives a high level of privacy protection at the expense of a reduced service offering.

Future research in the two-party case should include a more explicit determination of the extent of services that can be carried out in a privacy-preserving framework. This is particularly true for the database services part. A theoretical proof showing which part of the SQL algebra can be performed on encrypted data is still missing. One approach to investigate is whether the complete set of logical operations is sufficient to build the SQL algebra [Maurer, 2004]. Moreover, the practicality of the proposed service architecture should be analyzed in more complex settings and systems.

For the three-party case, we introduced the running example of a regional healthcare initiative that collects, analyzes and disseminates information about chronic disease treatment. We showed that a data snooper who uses mathematical programming techniques can derive tight bounds on confidential numerical values that were not included in the disseminated information. We proposed an iterative "audit & aggregate" methodology to detect and limit the privacy compromise called interval inference. A sample implementation showed that the data quality of the information that was finally disseminated with our methodology is higher than with comparable privacy protection methods such as random data perturbation. This accounts particularly for settings with a limited number of data holders like in the running healthcare example. We also gave experimental evidence of the fact that interval inference is more likely to occur in circumstances where "audit and aggregate" delivers better results and still has controllable complexity.

Future research in the three-party case embarks in four major directions. First, the complexity of the audit inhibits the use of "audit and aggregate" in large-size database systems. Some approaches exist with regard to linear programming problems, but further research would be beneficial for nonlinear programming problems that are induced e.g. by standard deviations in the disseminated marginal information. Second, disseminated information is not necessarily restricted to measures of centrality and dispersion, but can also extend to other measures such as partial or total order that might help patients to rank for instance the performance of HMOs. The integration of such information is theoretically possible, but experimental evidence about characteristics and sensitivities can only be obtained via a specific implementation. We also assume that our approach is suited for higher-dimensional data than in the analyzed two-dimensional setting, but a specific implementation would yield evidence about practical particularities. The third research direction is the design of a service provider or, more specifically, a mediator that does not necessarily have to be trusted. Cryptography yields some promising approaches

[Goldreich, 1998] that need further research regarding their applicability in mediator-based information systems. The fourth research direction is the integration of "audit and aggregate" into practical systems that are concerned with all kinds of privacy compromises. In healthcare for example, such a system would include the anonymization of individual patient records and the removal of sensitive attributes in queries of unauthorized users. It should be investigated how "audit and aggregate" can be integrated into complex systems where interval inference plays an important role.

The missing awareness of privacy issues and the lack of technical opportunities to balance privacy still inhibit electronic commerce. Many major web sites such as online book stores still do not offer even the most obvious options (e.g. to simply turn personalization services off). We think that the integration of trade-off techniques, independent from their sophistication, can help promote the continuous development of electronic commerce. With regard to public policy, a conflict often exists between each citizen's privacy and higher societal aims such as national security. Legislation has to trade off these competing interests and has to ensure that neither the citizens' privacy is compromised nor the activities of the authorities are hampered completely.

To summarize, this thesis shows that it is technically possible to trade off data privacy and data utility in web-based services. For many services, the quality of the service result increases with an extension of the provided personal input data. We show for the specific service of health data dissemination that for a given a level of privacy protection by the data holders, we can automatically generate a service result, i.e. the health report, with a high data utility. Going beyond the technical methods proposed in this thesis, we think it is worth analyzing the correlation between the technical trade-off opportunities and the actual behavior of the data holders. The question is whether the techniques suggested in this thesis can contribute to the reduction of extreme usage behavior that [Ackerman, et al., 1999] call "privacy fundamentalism" and "marginal concern".

References

Ackerman, M.; Cranor, L. and Reagle, J. (1999): Privacy in e-commerce: examining user scenarios and privacy preferences, Proceedings of the 1st ACM conference on Electronic commerce (ACM-EC '99), Denver, Colorado. URL: http://portal.acm.org/citation.cfm?id=336992.336995&coll=portal&dl=ACM&type=series&idx=336992&part=Proceedings&WantType=Proceedings&title=Electronic%20Commerce

Adam, N. and Wortman, J. (1989): Security-control methods for statistical databases, ACM Computing Surveys (vol. 21), No. 4, pp. 515-556. URL: http://portal.acm.org/citation.cfm?id=76895&dl=ACM

Agrawal, R. and Srikant, R. (2000): Privacy-preserving Data Mining, Proceedings of the ACM Conference on Management of Data (SIGMOD 2000), Dallas, Texas. URL: http://portal.acm.org/citation.cfm?id=335438&dl=GUIDE&coll=GUIDE

Ahituv, N.; Lapid, Y. and Neumann, S. (1987): Processing encrypted data, Communications of the ACM (vol. 30), No. 9, pp. 777-780. URL: http://portal.acm.org/citation.cfm?id=30404&dl=ACM&coll=portal

Amazon.com (2004): Privacy Notice, 2004, May 26, April 3, 2003, http://www.amazon.com/exec/obidos/tg/browse/-/468496/103-7024010-9846235

Asonov, D. and Freytag, J. C. (2002): Almost Optimal Private Information Retrieval, Proceedings of the 2nd Workshop on Privacy Enhancing Technologies (PET '02), San Francisco. URL: http://www.springerlink.com/index/L74U3JTG48Y6EYT0.pdf

Asonov, D.; Freytag, J. C. and Schaal, M. (2001): Absolute Privacy in Voting, Proceedings of the Information Security Conference, Málaga, Spain. URL: http://www.isconference.org/cp.html

Ballou, D. P. and Tayi, G. (1999): Enhancing data quality in data warehouse environments, Communications of the ACM (vol. 42), No. 1, pp. 73-78. URL: http://portal.acm.org/citation.cfm?id=291471&dl=ACM&coll=portal

Barak, B.; Goldreich, O.; Impagliazzo, R.; Rudich, S.; Sahai, A.; Vadhan, S. and Yang, K. (2001): On the (Im)possibility of Obfuscating Programs, Proceedings of Advances in Cryptology (CRYPTO 01). URL: http://www.springerlink.com/index/TELALQDCX3N600UF.pdf

Bayer, R. and McCreight, E. M. (1972): Organization and Maintenance of Large Ordered Indices, Acta Informatica, No. 1, pp. 173-189.

Berendt, B. and Teltzrow, M. (2003): Adressing Users' Privacy Concerns for Improving Personalization Quality: Towards an Integration of User Studies and Algorithm Evaluation, Proceedings of the Workshop on Intelligent Techniques for Web Personalization (ITWP '03), Acapulco, Mexico.

Berndt, D.; Fisher, J.; Hevner, A. and Studenicki, J. (2001): Healthcare Data Warehousing and Quality Assurance, IEEE Computer (vol. 34), No. 12, pp. 56-65. URL: http://www.coba.usf.edu/berndt/research/papers/computer2001quality.pdf

Boyens, C. and Fischmann, M. (2003): Profiting from Untrusted Parties in in Web-based Applications, Proceedings of the 4th International Conference on Electronic Commerce and Web Technologies (EC-Web '03), Prague, CZ. URL: http://www.springerlink.com/index/W0DEAY7BQ59TPQ2A.pdf

Boyens, C. and Günther, O. (2002): Trust Is not Enough: Privacy and Security in ASP and Web Service Environments, Proceedings of the 6th East-European Conference on Advances in Database and Information Systems (ADBIS 2002), Bratislava, Slovakia. URL: http://portal.acm.org/citation.cfm?id=676750&dl=ACM&coll=portal

Boyens, C. and Günther, O. (2003): Using Online Services in Untrusted Environments - A Privacy-Preserving Architecture, Proceedings of the 11th European Conference on Information Systems (ECIS '03), Naples, Italy. URL: http://www.wiwi.hu-berlin.de/~boyens/pubs/ECIS-Boyens_Guenther-Using_Online_Services.pdf

Boyens, C. and Günther, O. (2004): Detection and limitation of interval inference in statistical databases, Poster Session at the 16th International Conference on Scientific and Statistical Database Management (SSDBM 2004), Santorini, Greece. URL: http://cgi.di.uoa.gr/~ssdbm04/inf1.htm

Boyens, C.; Günther, O. and Teltzrow, M. (2002): Privacy Conflicts in CRM Services for Online Shops: A Case Study, Proceedings of the IEEE ICDM Workshop on Privacy, Security and Data Mining, Maebashi City, Japan. URL: http://portal.acm.org/citation.cfm?id=850787&dl=ACM&coll=portal

Boyens, C.; Krishnan, R. and Padman, R. (2004): On Privacy-Preserving Access to Distributed Heterogeneous Healthcare Data, Proceedings of the 37th Hawai'ian International Conference on System Sciences (HICSS-37), Hawai'i. URL: http://www.sigmod.org/sigmod/dblp/db/conf/hicss/hicss2004-6.html

Boyens, C. and Padman, R. (2003): On the Design of Data Dissemination Strategies in the Presence of Interval Inference, Proceedings of the 13th Workshop on Information Technology and Systems (WITS '03), Seattle, WA. URL: http://www.wiwi.hu-berlin.de/~boyens/pubs/WITS-Boyens_Padman-On_the_Design.pdf

Brackstone, G. J. (1999): Managing data quality in a statistical agency, Survey methodology (vol. 25), No. 02. URL: http://www.statcan.ca/english/IPS/Data/12-001-XPB19990024877.htm

Brickell, E. and Yacobi, Y. (1987): On privacy homomorphisms, Proceedings of Advances in Cryptology (EUROCRYPT '87). URL: http://dsns.csie.nctu.edu.tw/research/crypto/HTML/PDF/E87/117.PDF

Canny, J. (2002a): Collaborative Filtering with privacy, Proceedings of the 2002 IEEE Symposium on Security and Privacy. URL: http://portal.acm.org/citation.cfm?id=830525&dl=ACM&coll=GUIDE

Canny, J. (2002b): Collaborative Filtering with privacy via factor analysis, Proceedings of the 25th Annual International ACM Conference on Research and Development in Information Retrieval (SIGIR 2002), Tampere, Finland. URL: http://portal.acm.org/citation.cfm?id=564419&dl=ACM&coll=portal

Carter, T. (2000): ASP arena: Security remains top concern, ASPstreet.com, October 31, 2000, http://www.aspstreet.com/archive/d.taf/what%2Cshow/id%2C4581/c%2C22

Chaudhuri, S. and Dayal, U. (1997): An Overview of Data Warehousing and OLAP, ACM SIGMOD Record (vol. 26), No. 1, pp. 65-74. URL: http://www.acm.org/sigmod/record/issues/9703/chaudhuri.ps

Chin, F. Y. and Özsoyoglu, G. (1982): Auditing and inference control in statistical databases, IEEE Transactions on Software Engineering (vol. 8), No. 6, pp. 113-139.

Chor, B.; Kushilevitz, E.; Goldreich, O. and Sudan, M. (1995): Private Information Retrieval, Journal of the ACM (vol. 45), No. 6, pp. 965-981. URL: http://portal.acm.org/citation.cfm?id=293350&dl=ACM&coll=portal

Chowdhury, S. D.; Duncan, G.; Krishnan, R.; Roehrig, S. and Mukherjee, S. (1999): Disclosure detection in multivariate categorical databases: auditing confidentiality protection through two new matrix operators, Management Science (vol. 45), No. 12, pp. 1710-1723. URL: http://portal.acm.org/citation.cfm?id=337139&jmp=indexterms&dl=portal&dl=ACM

Clifton, C. (2001): Privacy Preserving Distributed Data Mining, Purdue University, Department of Computer Sciences, November 9, 2001, http://www.cs.purdue.edu/homes/clifton/DistDM/CliftonDDM.pdf

Clifton, C. and Marks, D. (1996): Security and privacy implications of data mining, ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery. URL: http://www.cs.purdue.edu/homes/clifton/document/dmkd.pdf

Codd, E. F. (1970): A Relational Model of Data for Large Shared Data Banks, Communications of the ACM (vol. 13), No. 6, pp. 377-387. URL: http://portal.acm.org/citation.cfm?id=362685&dl=ACM&coll=portal

Cox, L. H. (1980): Suppression methodology and statistical disclosure control, Journal of the Amercian Statistical Association (vol. 75), No. 370, pp. 377-385.

CSI (2003): CSI/FBI Computer Crime and Security Survey, Computer Security Issues and Trends (vol. VIII), No. 1. URL: http://www.gocsi.com/awareness/topline.jhtml

Cutler, M. and Sterne, J. (2000): E-Metrics. Business Metrics For The New Economy, NetGenesis, http://www.emetrics.org/articles/emetrics.pdf

Damiani, E.; Vimercati, S. De Capitani di; Jajodia, S.; Paraboschi, S. and Samarati, P. (2003): Balancing Confidentiality and Efficiency in Untrusted Relational DBMSs, Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS 2003), Washington, DC, USA. URL: http://portal.acm.org/citation.cfm?id=948124

DARPA (1981a): RFC 791: Internet Protocol, Defense Advanced Research Projects Agency, September 1981, http://www.faqs.org/rfcs/rfc791.html

DARPA (1981b): RFC 793: Transmission Control Protocol, Defense Advanced Research Projects Agency, September 1981, http://www.faqs.org/rfcs/rfc793.html

Denning, D. (1982): Cryptography and Data Security, Addison Wesley.

Denning, D. E. (1980): Secure statistical databases with random sample queries, ACM Transactions on Database Systems (TODS) (vol. 5), No. 3, pp. 291-315. URL: http://portal.acm.org/citation.cfm?id=320616&coll=portal&dl=ACM

DHS (2004): Office of Homeland Security, 2004, July 6, http://www.whitehouse.gov/homeland

Dobkin, D.; Jones, A. K. and Lipton, R. J. (1979): Secure databases: Protection against user influence, ACM Transactions on Database Systems (TODS) (vol. 4), No. 1, pp. 97-106. URL: http://portal.acm.org/citation.cfm?id=320068&dl=ACM&coll=portal

Domingo-Ferrer, J. (1996): A new privacy homomorphism and applications, Information Processing Letters (vol. 60), No. 5, pp. 277-282.

Domingo-Ferrer, J. (1997): Multi-application smart cards and encrypted data processing, Future Generation Computer Systems (vol. 13), No. 1, pp. 65-75. URL: http://citeseer.ist.psu.edu/299517.html

Domingo-Ferrer, J. and Herrera-Joancomarti, J. (1999): A privacy homomorphism allowing field operations on encrypted data, Jornades de Matematica Discreta i Algorismica, Barcelona.

Domingo-Ferrer, J. and Mateo-Sanz, J. M. (2002): Practical Data-Oriented Microaggregation for Statistical Disclosure Control, IEEE Transactions on Knowledge and Data Engineering (vol. 14), No. 1, pp. 189-201. URL: http://portal.acm.org/citation.cfm?id=628201&dl=ACM&coll=portal

Domingo-Ferrer, J.; Mateo-Sanz, J. M. and Torra, V. (2001): Comparing SDC methods for microdata on the basis of information loss and disclosure risk, Pre-proceedings of ETK-NTTS '2001 (vol. 2), Luxemburg.

Domingo-Ferrer, J.; Oganian, A. and Torra, V. (2002): Information-theoretic disclosure risk measures in statistical disclosure control of tabular data, Proceedings of the 14th International Conference on Scientific and Statistical Database Management (SSDBM '02), Edinburgh, Scotland. URL: http://ssdbm.soc.napier.ac.uk/show.php3?op=show&page=finalprogramme

Duncan, G.; Fienberg, S.; Krishnan, R.; Padman, R. and Roehrig, S. (2001a): Disclosure Limitation Methods and Information Loss for Tabular Data, Doyle, P.; Lane, J.; Theeuwes, J. and Zayatz, L., Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies pp. 135-166, Elsevier. URL: http://www.niss.org/dg/technicalreports.html

Duncan, G.; Keller-McNulty, S. and Stokes, S. (2001b): Disclosure Risk vs. Data Utility: The R-U Confidentiality Map, Technical Report LA-UR-01-6428, Los Alamos National Laboratory, http://www.niss.org/technicalreports/tr121.pdf

Duncan, G.; Krishnan, R.; Padman, R.; Reuter, P. and Roehrig, S. (2004): Exact and heuristic methods for cell suppression in multi-dimensional linked tables, Digital Government II: Technical reports, National Institute of Statistical Sciences, http://www.niss.org/dgii/TR/roehrig-suppres2.pdf

Economist (1999): The end of privacy: The surveillance society, The Economist, May 1, 1999, pp. 21-23.

EPIC (2004a): Comments of the Electronic Privacy Information Center, Electronic Privacy Information Center, July 1, 2004, http://www.epic.org/privacy/airtravel/rt_comments.pdf

EPIC (2004b): Cookies, Electronic Privacy Information Center, May 19, 2004, http://www.epic.org/privacy/internet/cookies/

Ernst & Young (2003): P3P Dashboard Report, July 2003, www.ey.com/global/download.nsf/US/P3P_Dashboard_-_July_2003/$file/E&YP3PDashboardJuly2003.pdf

EU (1995): Directive 95/46/EC of the European Parliament and of the European Council on the protection of individuals with regard to the processing of personal data and on the free movement of such data, European Union, http://europa.eu.int/comm/internal_market/en/media/dataprot/law/index.htm

EU (2002): Directive 2002/58/EC of the European Parliament and of the European Council Concerning the Processing of Personal Data and the Protection of Privacy in the Electronic Communications Sector, European Union, July 12, 2002, http://europa.eu.int/eur-lex/pri/en/oj/dat/2002/l_201/l_20120020731en00370047.pdf

Fellegi, I. P. (1972): On the question of statistical confidentiality, Journal of the American Statistical Association (vol. 67), No. 337, pp. 7-18.

Fieger, W. (1996): Mathematik für Wirtschaftswissenschaftler, Universität Karlsruhe, Skript zur Vorlesung WS 1996/97

Fischmann, Matthias and Günther, Oliver (2003): Privacy Tradeoffs in Database Service Architectures, Proceedings of the First ACM International Workshop on Business-Driven Security (BIZSEC 2003), Fairfax, VA. URL: http://www.acm.org/sigs/sigsac/ccs/CCS2003/bizsec.html

Fourer, R.; Gay, D. M. and Kernighan, B. W. (2003): AMPL - A modelling language for mathematical programming, 2. ed., Thomson.

Fox, S. and Rainie, L. (2000): Trust and Privacy Online: Why Americans Want to Rewrite the Rules, Pew Internet & American Life Project, Washington, D.C., http://www.pewinternet.org/report_display.asp?r=19

Goldreich, O. (1998): Secure Multi-Party Computation, http://www.wisdom.weizmann.ac.il/~oded/pp.html

Gopal, R.; Goes, P. and Garfinkel, R. (1998): Interval Protection of Confidential Information in a database, INFORMS Journal on Computing (vol. 10), No. 3.

Hacigumus, H.; Iyer, B. and Mehrota, S. (2002a): Providing Database as a Service, Proceedings of the IEEE International Conference on Data Engineering (ICDE 2002), San Jose, CA.

Hacigumus, H.; Mehrotra, S.; Iyer, B. and Li, C. (2002b): Executing SQL over Encrypted Data in the Database Service Provider Model, Proceedings of the ACM Conference on Management of Data (SIGMOD 2002).

Harangsri, B.; Matsushima, S.; Shepherd, J. and Ngu, A. H. H. (1997): Handling missing values in database systems using a naive bayesian classifier, Proceedings of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, Tucson, Arizona.

HIPAA (1996): Healthcare Insurance Portability Accountability Act, http://www.cms.hhs.gov/hipaa/

IAO (2003): Report to the United States Congress regarding the Total Information Awareness (TIA) program, DARPA Information Awareness Office, May 20, 2003, http://www.epic.org/privacy/profiling/tia/may03_report.pdf

IBM (1999): IBM Multi-national Consumer Privacy survey, 2004, July 8, http://www-1.ibm.com/services/files/privacy_survey_oct991.pdf

IDC (1999): The ASP's impact on the IT industry: An IDC-wide opinion, International Data Corporation, www.idc.com

Inmon, W. H. (1996): Building the Data Warehouse, 2. ed., John Wiley & Sons.

ITU (2000): Recommendation X.509, International Telecommunication Union, http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-X.509

Jarvenpaa, S. L.; Tractinsky, N. and Vitale, M. (2000): Consumer Trust in an Internet Store, Information Technology and Management (vol. 1), No. 1-2, pp. 45-71.

Johnson, R.; Molnar, D.; Song, D. X. and Wagner, D. (2002): Homomorphic Signature Schemes, Proceedings of CT-RSA 2002, San Jose, CA.

Kobsa, A. (2001): Tayloring Privacy to Users' Needs, Proceedings of the 8th International Conference on User Modeling, Sonthofen, Germany.

Kobsa, A. (2002): Personalized Hypermedia and International Privacy, Communications of the ACM (vol. 45), No. 5, pp. 64-67.

Kossmann, D. (2000): The state of the art in distributed query processing, ACM Computing Surveys (vol. 32), No. 4, pp. 422-469.

Krishnan, R.; Li, X.; Steier, D. and Zhao, L. (2001): On Heterogeneous Database Retrieval: A Cognitively Guided Approach, Information Systems Research (vol. 12), No. 3, pp. 286-301. URL: http://www.extenza-eps.com/extenza/contentviewing/viewArticle.do?articleId=9711&objectIDValue=9711&type=article

Lenz, H.-J. and Rödel, E. (1991): Statistical Quality Control of Data, Proceedings of DGOR.

Li, Y.; Wang, L. and Jajodia, S. (2002a): Preventing Interval-Based Inference by Random Data Perturbation, Proceedings of the Second International Workshop on Privacy Enhancing Technologies (PET 2002), San Francisco, CA.

Li, Y.; Wang, L.; Wang, X. and Jajodia, S. (2002b): Auditing Interval-Based Inference, Proceedings of the 14th Conference on Advanced Information Systems

Engineering (CAiSE'02), Toronto, Canada.

Li, Y.; Zhu, S.; Wang, L. and Jajodia, S. (2002c): A privacy-enhanced microaggregation method, Proceedings of Foundations of Information and Knowledge Systems (FoIKS 2002), Salzan Castle, Germany.

Lindell, Y. and Pinkas, B. (2000): Privacy-preserving data mining, Proceedings of Advances in Cryptology (CRYPTO 2000), Santa Barbara, CA.

Maurer, U. (2004): The Role of Crypthography in Database Security, Proceedings of the ACM Conference on Management of Data (SIGMOD '04), Paris, France.

Mizoras, A.; Whalen, M.; Goepfert, J.; Moser, K. and Graham, S. (2001): Worldwide ASP revenues Approached $1 Billion in 2000, IDC Bulletin, International Data Corporation, June 5, 2001

Naor, Moni and Pinkas, Benny (2001): Efficient Oblivious Transfer Protocols, Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms.

Naumann, F. (2002): Quality-driven query answering for integrated information systems (vol. 2261), Lecture Notes in Computer Science, Springer Verlag.

Netscape (1996): SSL 3.0 Specification, November 1996, http://wp.netscape.com/eng/ssl3/

Neumann, B. C. and Ts'o, T. (1994): Kerberos: An Authentication Service for Computer Networks, IEEE Communications (vol. 32), No. 9, pp. 33-38. URL: http://gost.isi.edu/publications/kerberos-neuman-tso.html

Orwell, G. (1949): Nineteen Eighty-Four, Chelsea House Publishers.

Ozsoyoglu, G.; Singer, D. and Chung, S. (2003): Anti-Tamper Databases: Querying Encrypted Databases, Proceedings of the 17th Annual IFIP WG 11.3 Working Conference on Database and Applications Security, Estes Park, Colorado.

P3P (2004): The Platform for Privacy Preferences, 2004, July 8, http://www.w3.org/P3P/

PATRIOT (2001): Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism (PATRIOT) Act, H.R.3162 in the United States Senate. URL: http://www.epic.org/privacy/terrorism/hr3162.html

PHC4 (2002): Diabetes Hospitalization Report, Pittsburgh, PA, Pittsburgh Healthcare Cost Containment Council, Nov. 2002, www.phc4.org/adobe/Diab01.pdf

Rezgui, A.; Ouzzani, M.; Bouguettya, A. and Medjahed, B. (2002): Preserving Privacy in Web Services, Proceedings of the Workshop on Web Information and Data Management (WIDM '02), McLean, VA. URL: http://www.se.cuhk.edu.hk/~eplim/widm2002/

Rindfleisch, T. C. (1997): Privacy, Information Technology and Healthcare, Communications of the ACM (vol. 40), No. 8, pp. 92-100.

Rivest, R.; Adleman, L. and Dertouzos, M. (1978a): On Data Banks and Privacy Homomorphisms, Foundations of Secure Computation, Academic Press, New York.

Rivest, R.; Shamir, A. and Adleman, L. (1978b): A Method for Obtaining Digital Signatures and Public Key Cryptosystems, Communications of the ACM (vol. 21), No. 2, pp. 120-126. URL: http://portal.acm.org/citation.cfm?id=359340.359342&coll=portal&dl=ACM&idx=359340&part=periodical&WantType=periodical&title=Communications%20of%20the%20ACM&CFID=25632330&CFTOKEN=20695504

Rusch, J. J. (2004): The "Social Engineering" of Internet Fraud, United States Department of Justice, 2004, July 10, http://www.isoc.org/isoc/conferences/inet/99/proceedings/3g/3g_2.htm

Schily, O. (2004): Wir leben in Zeiten epochaler Bedrohung, Interview in Frankfurter Allgemeine Sonntagszeitung (March 21, 2004).

Schneier, B. (1996): Applied Crypthography, 2. ed., John Wiley & Sons.

Shannon, C. (1948): A mathematical theory of communication, Bell System Technical

Journal (vol. 27), pp. 379-423.

Shoshani, A. (1982): Statistical databases: Characteristics, problems and some solutions, Proceedings of the 8th International Conference on Very Large Databases (VLDB '82), Bombay, India.

Shoshani, A. (1997): OLAP and statistical databases: similarities and differences, Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems (PODS '97), Tucson, Arizona. URL: http://classweb.gmu.edu/kersch/inft864/Readings/Shoshani/olap.vs.sdb.paper.PODS97.pdf

Smith, S. W. and Weingart, S. H. (1999): Building a High-Performance, Programmable Secure Coprocessor, Computer Networks, Special Issue on Computer Network Security (vol. 31), No. 9, pp. 831-860. URL: http://portal.acm.org/citation.cfm?id=324128&dl=ACM&coll=portal

Song, Dawn Xiaodong; Wagner, David and Perrig, Adrian (2000): Practical Techniques for Searches on Encrypted Data, Proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA.

Spiekermann, S.; Grossklags, J. and Berendt, B. (2001): E-privacy in 2nd generation E-commerce: privacy preferences versus actual behavior, Proceedings of the 3rd ACM conference on Electronic Commerce (ACM-EC '01), Tampa, FL.

Stallings, W. (1999): Cryptography and Network Security: Principles and Practice, Prentice Hall.

Sweeney, L. (2001): Computational Disclosure Control: A Primer on Data Privacy Protection, Massachusetts Institute of Technology.

Sweeney, L. (2002a): Achieving k-Anonymity Privacy Protection Using Generalization and Suppression, International Journal on Uncertainty, Fuzziness and Knowledge-based Systems (vol. 10), No. 5, pp. 571-588.

Sweeney, L. (2002b): k-anonymity: A model for protecting privacy, International Journal

on Uncertainty, Fuzziness and Knowledge-based Systems (vol. 10), No. 5, pp. 557-570.

Tamm, G. (2003): Netzbasierte Dienste - Angebot, Nachfrage und Matching, Dissertation, Humboldt Universität zu Berlin. URL: http://edoc.hu-berlin.de/abstract.php3/dissertationen/tamm-gerrit-2003-05-09

Tamm, G. and Günther, O. (2004): Webbasierte Dienste: Technologien, Märkte und Geschäftsmodelle, Springer Verlag.

Teltzrow, M. and Kobsa, A. (2004): Impact of user privacy preferences on personalized systems, Karat, C.-M.; Blom, J. and Karat, J., Designing Personalized User Experiences for eCommerce, Kluwer Academic Publishers, Dordrecht, Netherlands.

Terdimann, R.; Apfel, A.; Paulak, E. and Berg, T. (2000): How hard will ASPs bite the IT Industry, Strategic Analysis Report Nr. R-12-6618, Garnter Group, Inc.

Time (1997): The death of privacy, Time Magazine, August 25, 1997.

Traub, J. F.; Yemini, Y. and Woznaikowski, H. (1984): The statistical security of a statistical database, ACM Transactions on Database Systems (TODS) (vol. 9), No. 4, pp. 672-679.

USPA (1974): The Privacy Act of 1974, 5 U.S.C. § 552a, United States Congress, http://www.usdoj.gov/foia/privstat.htm

Vaidya, J. and Clifton, C. (2002): A security mediator for health care information, Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (SIGKDD 2002), Edmonton, Alberta.

W3C (1997): Hypertext Transfer Protocol -- HTTP/1.1, World Wide Web Consortium, January 1997, http://www.w3.org/Protocols/rfc2068/rfc2068

W3C (2004): Web Services Architecture Requirements, World Wide Web Consortium, February 11, 2004, http://www.w3.org/TR/2004/NOTE-wsa-reqs-20040211/

Warren, S. and Brandeis, L. D. (1890): The right to privacy, Harvard Law Review (vol. IV), No. 5.

Wiederhold, G. (1993): Intelligent Integration of Information, Proceedings of the ACM Conference on Management of Data (SIGMOD 1993), Washington, D.C.

Wiederhold, G.; Bilello, M.; Sarathy, V. and Qian, X. (1996): A Security Mediator for Health Care Information, Proceedings of the 1996 AMIA Conference, Washington, D.C.

Willenborg, L. and Waal, T. de (2001): Elements of Statistical Disclosure Control, Addison Wesley.

Winkler, W. (2002): Single ranking micro-aggregation and re-identification, Statistical Research Division report RR 2002/08, www.census.gov/srd/www/byyear.html

Winston, W. L. (1991): Introduction to mathematical programming: applications and algorithms, Duxbury Press.

Appendix

Appendix A: Data tables

Data tables for the implementation of the 2-party case

The 2-party implementation is described in Section 3.8.

	NO KEY	32 BIT	64 BIT	128 BIT
S_1: Average absence per dept. (ms)	88.2	92.684	95.404	97.762
Surcharge for encryption	0%	5%	8%	11%

	NO KEY	32 BIT	64 BIT	128 BIT
Creation Time (sec)	9.111	11.818	13.199	19.35
Surcharge for encryption	0%	30%	45%	112%
Table size (KB)	84	120	208	232
Surcharge for encryption	0%	43%	148%	176%

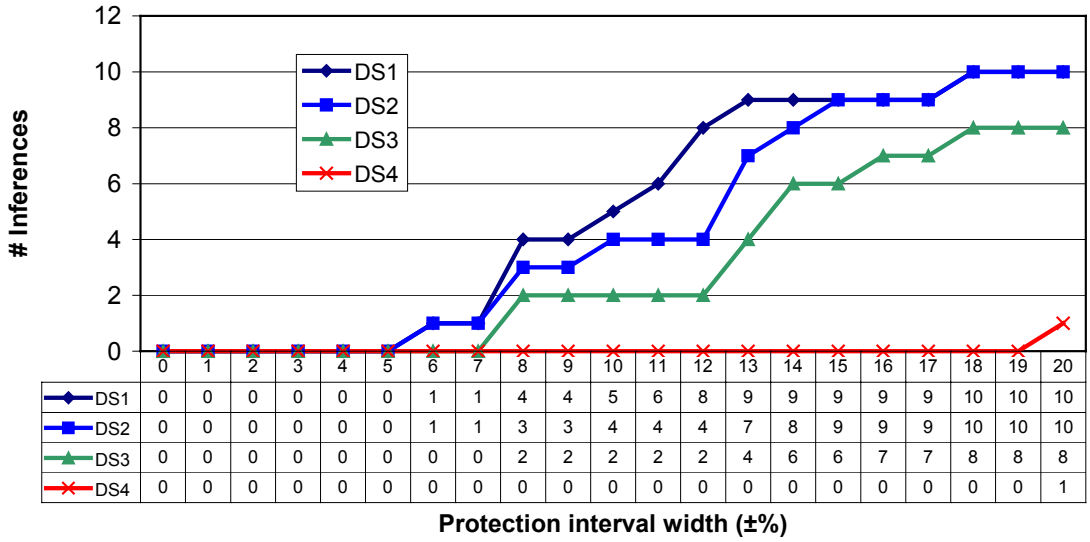
Table 0-1: Creation times and disk space for the unencrypted and the encrypted employee table

	NO KEY	32 BIT	64 BIT	128 BIT
Creation Time (sec)	9.405	11.951	28.236	51.139
Surcharge for encryption	0%	27%	200%	444%
Table size (KB)	80	208	348	456
Surcharge for encryption	0%	160%	335%	470%

Table 0-2: Creation times and disk space for the unencrypted and the encrypted monthly_account table

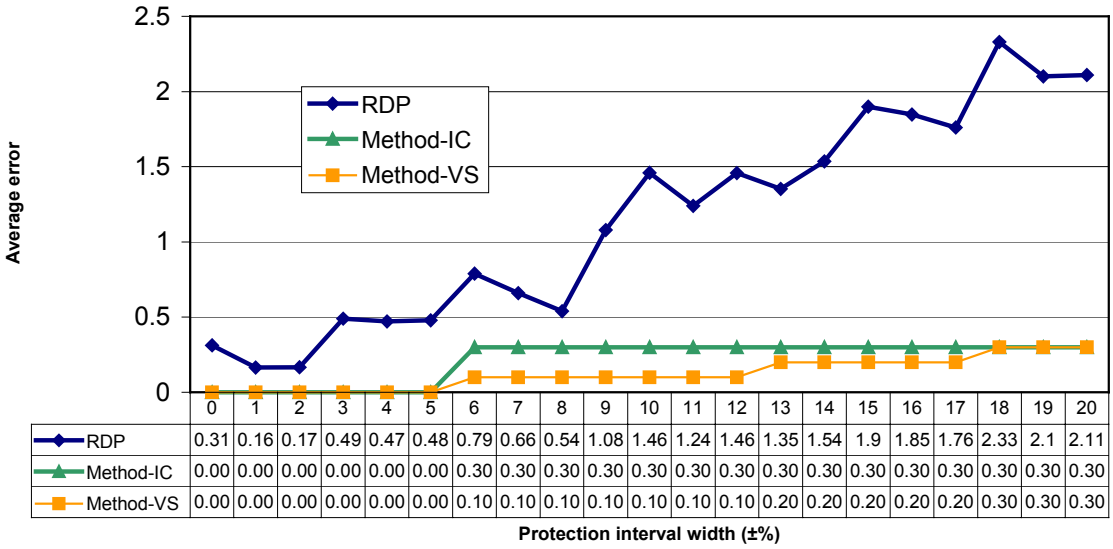
Data tables for the implementation of the 3-party case

Figure 4-16: Dissemination strategies and inferred intervals for different protection policies:



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DS1	0	0	0	0	0	0	1	1	4	4	5	6	8	9	9	9	9	9	10	10	10
DS2	0	0	0	0	0	0	1	1	3	3	4	4	4	7	8	9	9	9	10	10	10
DS3	0	0	0	0	0	0	0	0	2	2	2	2	2	4	6	6	7	7	8	8	8
DS4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

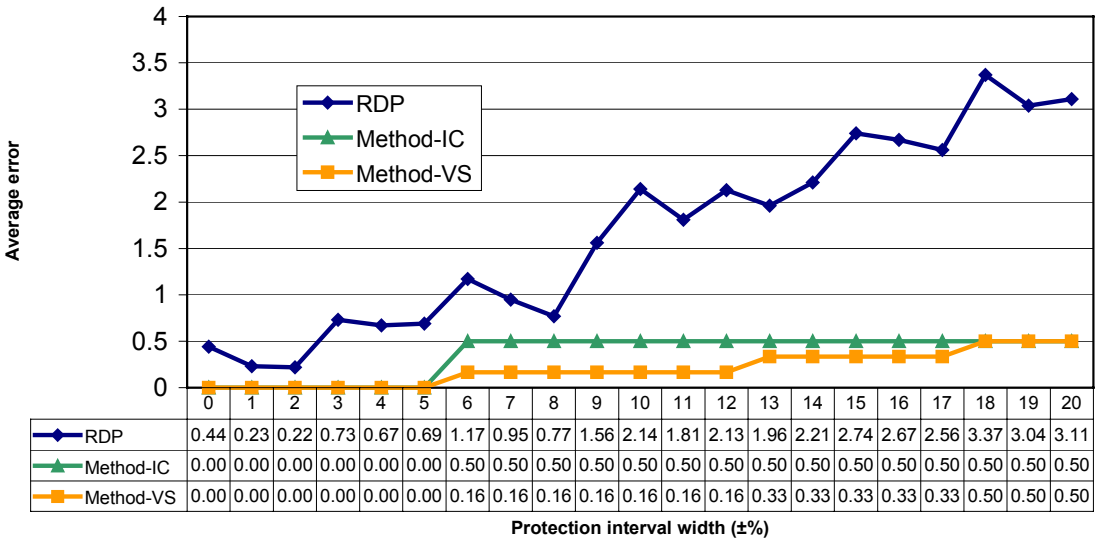
Figure 4-17: Total average relative error (TARE)



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
RDP	0.31	0.16	0.17	0.49	0.47	0.48	0.79	0.66	0.54	1.08	1.46	1.24	1.46	1.35	1.54	1.9	1.85	1.76	2.33	2.1	2.11
Method-IC	0.00	0.00	0.00	0.00	0.00	0.00	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30
Method-VS	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.20	0.20	0.20	0.20	0.20	0.30	0.30	0.30

Protection interval width (±%)

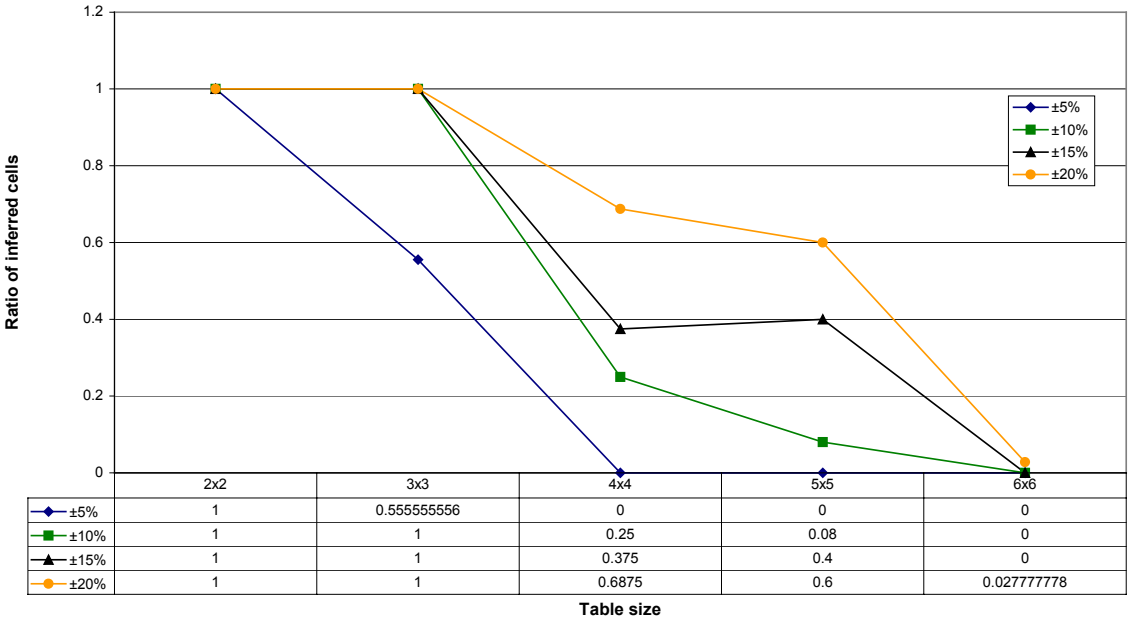
Figure 4-18: Average relative column error (ARE_{col})



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
RDP	0.44	0.23	0.22	0.73	0.67	0.69	1.17	0.95	0.77	1.56	2.14	1.81	2.13	1.96	2.21	2.74	2.67	2.56	3.37	3.04	3.11
Method-IC	0.00	0.00	0.00	0.00	0.00	0.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
Method-VS	0.00	0.00	0.00	0.00	0.00	0.00	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.33	0.33	0.33	0.33	0.33	0.50	0.50	0.50

Protection interval width (±%)

Figure 4-19: Table size vs. ratio of inferred cells for different protection policies



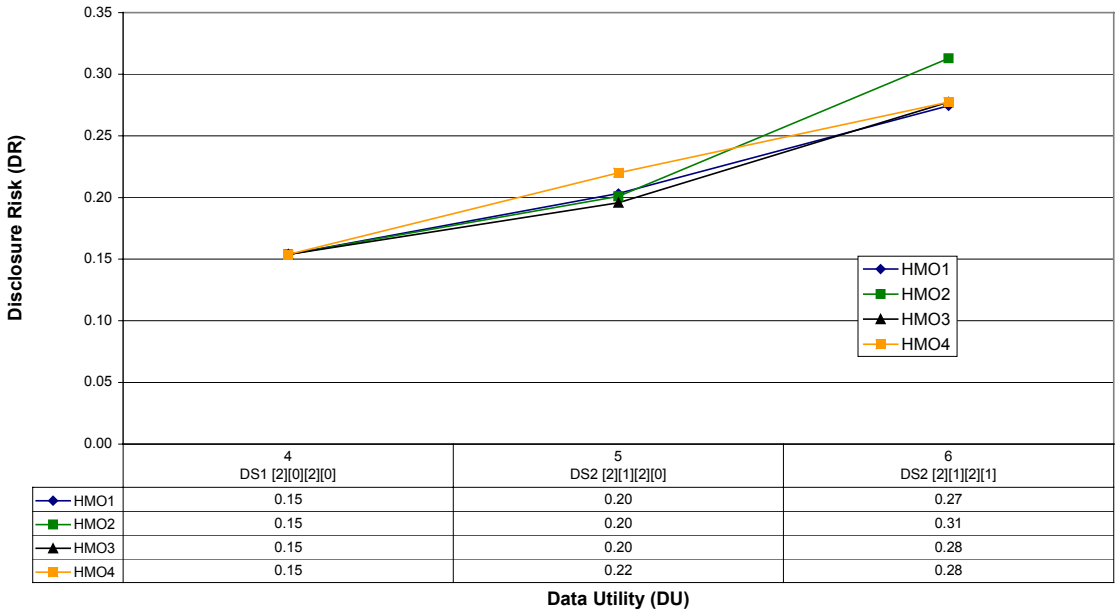
	2x2	3x3	4x4	5x5	6x6
±5%	1	0.555555556	0	0	0
±10%	1	1	0.25	0.08	0
±15%	1	1	0.375	0.4	0
±20%	1	1	0.6875	0.6	0.027777778

Table size

Data tables for the quantification of the privacy trade-off

The quantification of the privacy trade-off has been discussed in Section 5.1.3.

Figure 5-2: R-U confidentiality maps for all HMOs



	4 DS1 [2][0][2][0]	5 DS2 [2][1][2][0]	6 DS2 [2][1][2][1]
HMO1	0.15	0.20	0.27
HMO2	0.15	0.20	0.31
HMO3	0.15	0.20	0.28
HMO4	0.15	0.22	0.28

Appendix B: Java classes and methods

Java classes and methods for the implementation of the 2-party case

The implementation of the 2-party case is discussed in Section 3.8

CLASS PHTEST	
Class	PHActionListener implements ActionListener
Integer	calculateEntireResult
BigInteger	Euclid
JPanel	getPhpanel
void	initialize
void	setPhPanel
void	carryOutS1
void	createEmployeeTable
void	creatMonthlyAccountTable
void	dbAllOut
void	dbMonthlyAccountOut
void	dbOut
void	dbQuery
BigInteger	decrypt
BigInteger	encrypt
BigInteger	geta
BigInteger	getP
BigInteger	getQ
Boolean	standsFermatTest

Table 0-3: Methods in class PHTest

CLASS SERVICE PROVIDER	
BigInteger	getAverageDepartmentAbsence
BigInteger	getTotalWages

Table 0-4: Methods in class ServiceProvider

Java classes and methods for the implementation of the 3-party case

The implementation of the 3-party case is discussed in Section 4.8

CLASS MAIN	
void	main
void	createRawData
void	methodic
void	detectInference
double	edGaussian
void	setModel
void	calculateAvgShadowPrices
byte	countedInferences
void	methodVS
void	detectValueInference
void	calculateAvgValueShadowPrices

Table 0-5: Methods in class Main

Appendix C: AMPL Files

The use of AMPL to solve mathematical programming problems usually requires the following.

- An AMPL model file (*.mod).
- An AMPL data file (*.dat).

We used an AMPL script file (*.run) to read the data from a Microsoft Access database instead of writing the data in the AMPL data file directly.

TYPE OF FILE	FILE NAME
AMPL Model	multiple_constraints.mod
AMPL Script	read_data.run

Table 0-6: AMPL Files

The AMPL script file

read_data.run

```
#This file reads A,L,U from the Database, sets the model and
defines important parameters and options


reset;
option ampl_include '.\TABLES';
option display_1col 0;
option solution_precision 1e-2;
option display_precision 3;
option show_stats 0;
option solver_msg 0; #omits all messages issued by the solver


model ./MODELS/SSDBM/multiple_constraints.mod;


#read data from MS Access tables
table rows INOUT "ODBC" "TABLES/hicss.mdb" "rows": HMOS <- [HMOS],
row arithmetic mean   ~   arithmetic mean,   row mean interval   ~
```

```
mean_interval, row_standard_deviation ~ standard_deviation,
row_min_max_skew ~ min_max_skew;


table columns INOUT "ODBC" "TABLES/hicss.mdb" "columns": TESTS <-
[TESTS], column_arithmetic_mean ~ arithmetic_mean,
column_mean_interval ~ mean_interval, column_standard_deviation ~
standard_deviation, column_min_max_skew ~ min_max_skew;


table bounds INOUT "ODBC" "TABLES/hicss.mdb" "bounds": [HMOS,
TESTS], a, lower_bound ~ lb, upper_bound ~ ub;


read table rows;

read table columns;

read table bounds;


for {i in HMOS} {let I := i;} #"Import the No. of Rows from the
database table

for {j in TESTS} {let J := j;}
```

Figure 0-1: AMPL script file

The AMPL model file

multiple_constraints.mod

```
# AMPL Model for the Optimization Problems

param I>0;
param J>0;
param inference_flag;
set HMOS;# = 1..I;
set TESTS;# = 1..J;
set CATEGORIES:= 1..6;
#set PERCENTAGES within {HMOS, TESTS};

param row_arithmetic_mean {HMOS} >=0, <=1;
param row_mean_interval {HMOS} >=0, <=1;
param row_standard_deviation {HMOS} >=0;
param row_min_max_skew {HMOS} >=0, <=1;
param column_arithmetic_mean {TESTS} >=0, <=1;
param column_mean_interval {TESTS} >=0, <=1;
param column_standard_deviation {TESTS} >=0;
param column_min_max_skew {TESTS} >=0, <=1;
#param insider_row {TESTS} >=0, <=1;
param lower_bound {HMOS, TESTS} >=0, <=1; #those fixed by the HMO
DB administrator
param upper_bound {HMOS, TESTS} >=0, <=1;
param inferred_lb {HMOS, TESTS} >=0, <=1; #those inferred by the
snooper
param inferred_ub {HMOS, TESTS} >=0, <=1;
param interval_width {HMOS, TESTS} >=0;
param width_ratio {HMOS, TESTS};
param cons1dual{HMOS, TESTS}; #shows average shadow price for a
specific class of constraints (e.g.row_avg)
param cons2dual{HMOS, TESTS};
param cons3dual{HMOS, TESTS};
param park1min;
param park1max;
param park2min;
```

```
param park2max;

param park3min;

param park3max;

param count_inferences;

param data_utility{CATEGORIES} >=1, <=3;


var a {HMOS, TESTS} >=0.3, <=1;    # The percentages a_ij over HMOs
and Tests


minimize Lower_bound {i in HMOS, j in TESTS}: a[i,j];


maximize Upper_bound {i in HMOS, j in TESTS}: a[i,j];


# CATEGORY 1, DATA UTILTY 3

subject to Row_arithmetic_mean {i in HMOS}: # Row uppercase! -->
No confusion with param row_a...
    1/J * sum{j in TESTS} a[i,j] <= row_arithmetic_mean[i];


# CATEGORY 1, DATA UTILTY 2

subject to Row_mean_interval {i in HMOS}:
    row_mean_interval[i] <= 1/J * sum{j in TESTS} a[i,j] <=
row_mean_interval[i]+0.05;


# CATEGORY 2, DATA UTILTY 3

subject to Row_standard_deviation {i in HMOS}:
    1/J * sum{j in TESTS} (a[i,j] - 1/J * sum{k in TESTS}
a[i,k])^2 <= row_standard_deviation[i]^2;


# CATEGORY 2, DATA UTILTY 2

subject to Row_min_max_skew {i in HMOS}:
    max{j in TESTS} a[i,j] - min {j in TESTS} a[i,j] <=
row_min_max_skew[i];


# CATEGORY 4, DATA UTILTY 3

subject to Column_arithmetic_mean {j in TESTS}:
    1/I * sum{i in HMOS} a[i,j] = column_arithmetic_mean[j];


# CATEGORY 4, DATA UTILTY 2

subject to Column_mean_interval {j in TESTS}:
```

```
    column_mean_interval[j] <= 1/I * sum{i in HMOS} a[i,j] <=
column_mean_interval[j]+0.05;


# CATEGORY 5, DATA UTILTY 3
subject to Column_standard_deviation {j in TESTS}:
    1/I * sum{i in HMOS} (a[i,j] - 1/I * sum{k in HMOS} a[k,j])^2
<= column_standard_deviation[j]^2;


# CATEGORY 5, DATA UTILTY 2
subject to Column_min_max_skew {j in TESTS}:
    max{i in HMOS} a[i,j] - min {i in HMOS} a[i,j] <=
column_min_max_skew[j];


#subject to Insider_knowledge {j in TESTS}:
#   insider_row[j] = a[1,j];
```

Figure 0-2: AMPL model file

Appendix D: Screenshots

These screenshots are taken from the prototypical implementation in Section 4.8

The AMPL/Java Interface

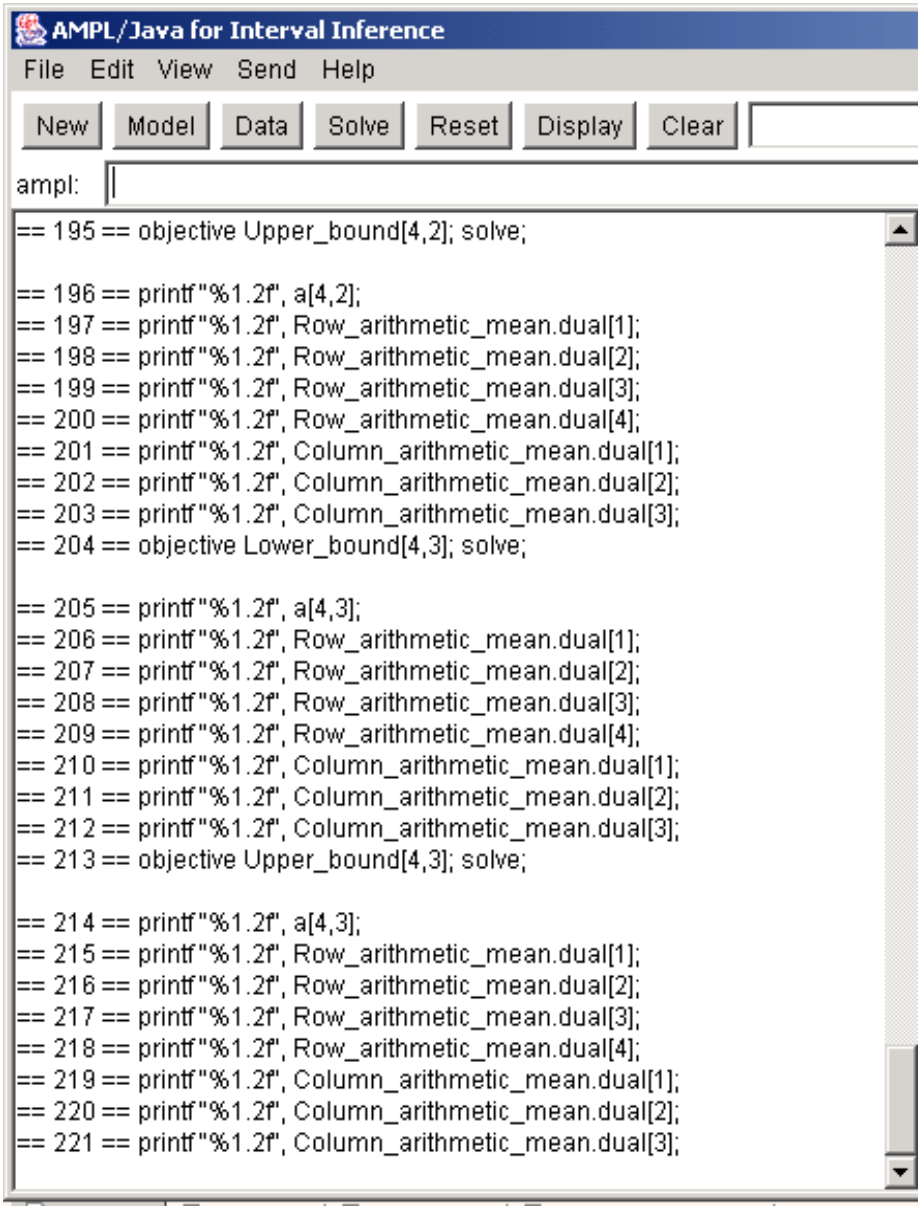


Figure 0-3: Screenshot from the adapted Java interface for AMPL

Screenshot for Method-IC

Method-IC was discussed in Section 4.6.

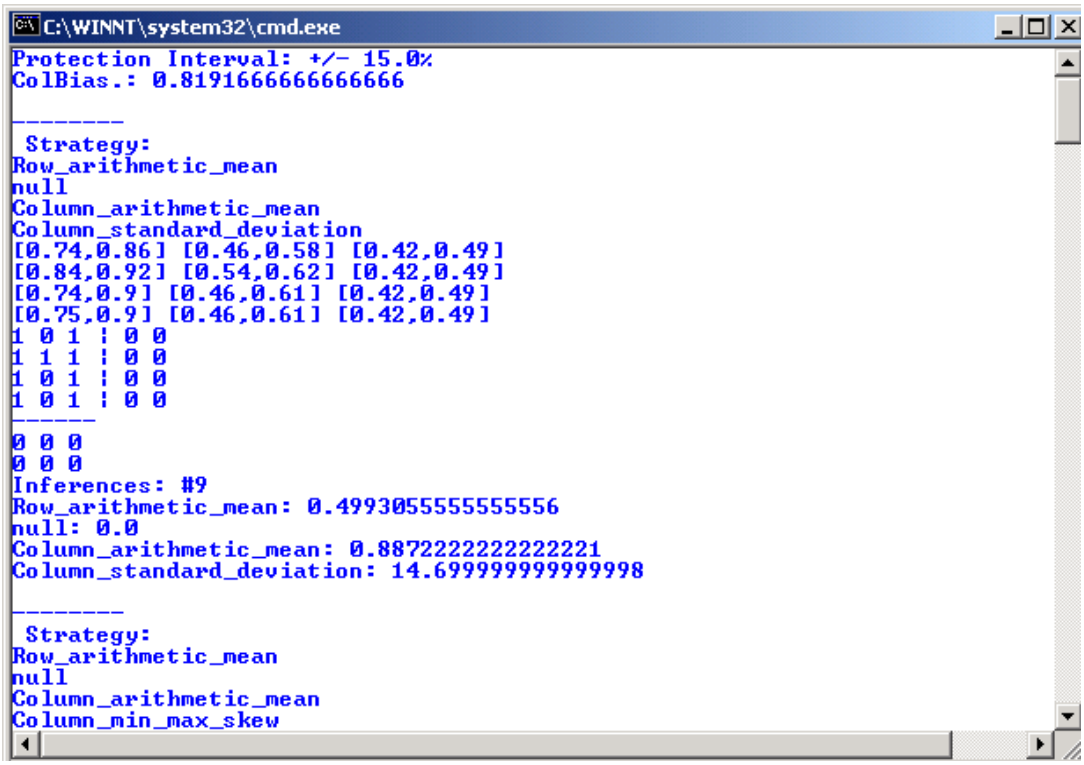


Figure 0-4: Screenshot of "audit and aggregate", Method-IC

Screenshot of Method-VS

Method-VS was discussed in Section 4.7.

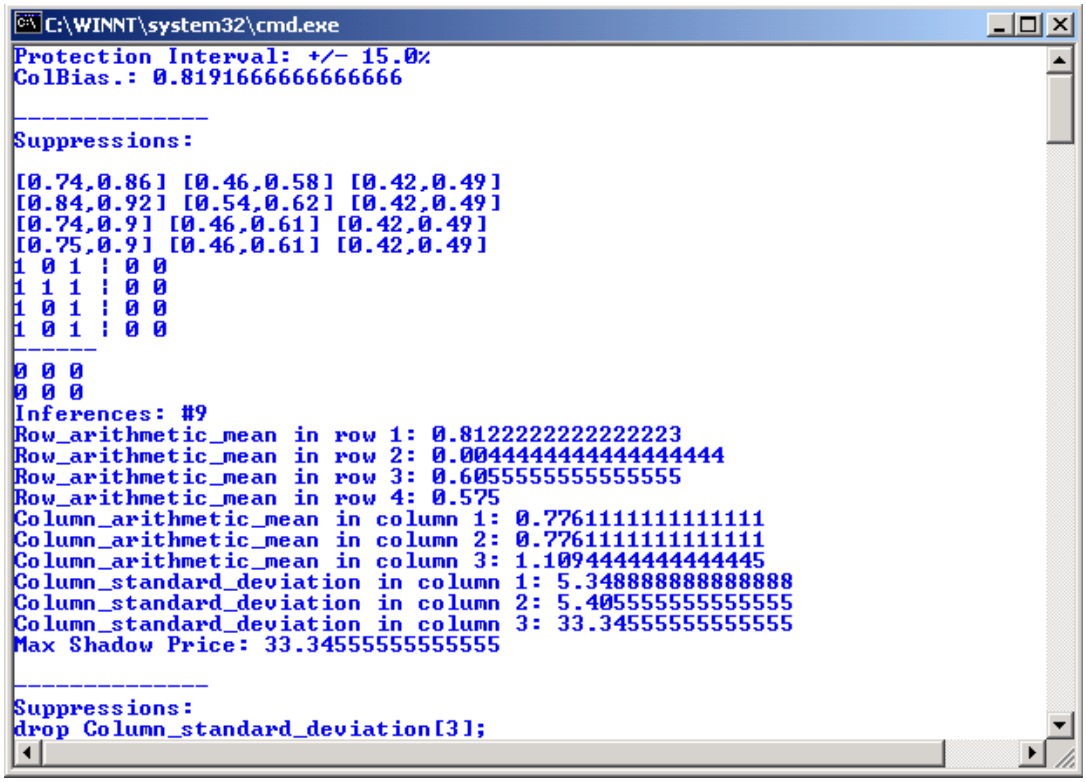


Figure 0-5: Screenshot of "audit and aggregate", Method-VS

Appendix E: Relational model for the 3-party case implementation

The relational database model for the 3-party case implementation was used in Section 4.8.

Relational model
bounds (HMOS, TESTS, a, lb, ub) **rows** (HMOS, arithmetic_mean, mean_interval, standard_deviation, min_max_skew) **columns** (TESTS, arithmetic_mean, mean_interval, standard_deviation, min_max_skew) **rdp_values** (HMOS, TESTS, a, lb, ub, a_rdp) **rdp_rows** (HMOS, arithmetic_mean, rdp_arithmetic_mean, rel_mean_error, standard_deviation, rdp_standard_deviation, rel_stdev_error) **rdp_columns** (TESTS, arithmetic_mean, rdp_arithmetic_mean, rel_mean_error, standard_deviation, rdp_standard_deviation, rel_stdev_error)

Figure 0-6: Relational model for the 3-party case

Underlined attributes indicate a primary key.

Empfangene Unterstützung und Hilfe durch Kollegen

- Prof. Rema Padman (Carnegie-Mellon University, USA) gab wertvolle Anregungen und Ideen für Kapitel 4.
- In Kapitel 3 sind Kommentare eines Reviewprozesses durch Gutachter der European Conference on Information Systems (ECIS 2003) eingeflossen.
- In Kapitel 3 sind Kommentare eines Reviewprozesses durch Gutachter der Conference on Electronic Commerce and Web Technologies (EC-Web '03) eingeflossen.
- In Kapitel 4 sind Kommentare eines Reviewprozesses durch Gutachter der Hawaii' International Conference on Systems Sciences (HICSS-37) eingeflossen.
- In Kapitel 4 sind Kommentare eines Reviewprozesses durch Gutachter des Workshop on Information Technology and Systems (WITS '03) eingeflossen.

Ich bezeuge durch meine Unterschrift, dass meine Angaben über die bei der Abfassung meiner Dissertation benutzten Hilfsmittel, über die mir zuteil gewordene Hilfe sowie über frühere Begutachtungen meiner Dissertation in jeder Hinsicht der Wahrheit entsprechen.

Berlin, 26.8.2004

Claus Boyens

Eidesstattliche Erklärung

Hiermit erkläre ich, Claus Boyens, dass ich mich bisher noch an keiner Institution einem Doktorexamen unterzogen habe. Ferner wurde die Dissertation bisher an noch keiner anderen Fakultät vorgelegt.

Berlin, 26.8.2004

Claus Boyens